

DataInterchange Administrator's Guide Version 3 Release 1

DataInterchange Administrator's Guide Version 3 Release 1

SB34-2002-04

Note!

Before using this information and the product it supports, be sure to read the general information under “Notices” on page xvii.

Fifth Edition (December 1998)

This book is a major revision and replaces SB34-2002-03. New editions will reflect changes in procedure or technical details.

This edition applies to Version 3 Release 1, Modification Level 0, of the DataInterchange for MVS licensed program number 5655-B29 and DataInterchange for CICS licensed program number 5655-B30, and to all subsequent releases and modifications until otherwise indicated in new editions. Use this publication only for the purposes stated in “To the Reader” on page xix.

Books are not stocked at the address given below. Request for copies should be made to your marketing representative.

A form for readers' comments is provided at the back of this publication. If the form has been removed, comments may be addressed to: IBM, Department RD10, P.O. Box 20143, Tampa, FL, 33633-0872, U.S.A.

You may use this form to communicate your comments about this publication with the understanding that IBM may use or distribute whatever information you supply in any way it believes appropriate without incurring any obligation to you.

© Copyright International Business Machines Corporation 1993, 1998. All rights reserved.

Note to U.S. Government Users — Documentation related to restricted rights — Use, duplication or disclosure is subject to restrictions set forth in GSA ADP Schedule Contract with IBM Corp.

Contents

Figures	xiii
Tables	xv
Notices	xvii
Trademarks and Service Marks	xvii
Softcopy Agreement	xvii
To the Reader	xix
Changes after Publication	xix
Who Should Read This Book	xix
Related Books	xix
What EDI Courses are Available	xx
How This Book Is Organized	xxi
Where to Find More Information	xxii
Information about Standards	xxii
Terms Used in This Book	xxiii
ANSI X12 and EDIFACT Terminology	xxiii
Syntax Conventions Used in This Book	xxiv
Understanding Syntax Diagrams	xxiv
Sending Comments for This Publication	xxvi
Summary of Changes	xxvii
Summary of Changes for DataInterchange Version 3 Release 1	xxvii
Summary of Changes for DataInterchange Version 2 Release 1	xxviii
Changes to the DataInterchange Host	xxix
Chapter 1. Introducing DataInterchange	1-1
Doing Business Through EDI	1-1
DataInterchange: An EDI Solution	1-2
Before You Begin	1-5
Setup for Sending and Receiving Data	1-5
Sending Your Data to a Trading Partner	1-6
Translating Application Data to Standard	1-6
Collecting Application Data for Translation	1-6
Translating Application Data as it is Generated	1-7
Enveloping and Sending Transactions	1-7
Envelope Overrides	1-7
Receiving Data From Your Trading Partner	1-8
Receiving and Deenveloping Transactions	1-8
Translating Standard Data to Application Format	1-9
Continuous Receive	1-9
Translating Application Data to Application Data	1-9
Functional Acknowledgments	1-9
Update Status	1-10
DataInterchange Reporting on Your EDI System	1-10
Online Inquiries Using the Transaction Store Facility	1-10
Formatted Reports Using the DataInterchange Utility	1-10
Transaction and Envelope Data Extracts Using the DataInterchange Utility	1-11
Management Reporting Data Extracts	1-11

Migrating Maps	1-12
Export/Import	1-12
Interactive Entry Facility (IEF)	1-12
Using DataInterchange with the DataInterchange Client Graphical User Interface	1-13
Client/Server Mode versus Standalone Mode	1-14
DataInterchange Client Databases	1-15
Naming Convention Changes for DataInterchange Client	1-15
Unavailable Functions in the DataInterchange Client Interface	1-15
Moving to DataInterchange Client	1-16
Moving Setup Profiles and Trading Partner Profiles	1-16
Moving Standards, Data Formats, and Maps	1-16
Chapter 2. Getting Started with DataInterchange	2-1
Accessing DataInterchange for MVS	2-1
Logon Options for DataInterchange for MVS	2-1
Logging On to DataInterchange for MVS	2-3
Accessing DataInterchange for CICS	2-3
Logon Options for DataInterchange for CICS	2-3
Logging On to DataInterchange for CICS	2-3
Selecting from the Menu	2-5
Using DataInterchange Panels	2-6
Panel Areas	2-6
Hierarchy of User Actions	2-8
Entering Commands at the Command Prompt	2-8
Entering Data	2-9
Initial Cursor Position	2-9
Requesting Help (F1)	2-9
Scrolling and Using the List Action	2-10
Printing Reports and Lists	2-11
Exit (F3) and Cancel (F12)	2-11
Prompt (F4)	2-11
Extended Prompt for Mapping and Application Data Formats	2-14
Chapter 3. Importing and Requesting EDI Standards Electronically	3-1
Importing EDI and Envelope Standards	3-1
Importing New Standards	3-1
Obtaining Standards through Electronic Distribution	3-2
Program Prerequisites	3-2
Requesting New Standards	3-3
Allocating Your Data Sets	3-6
DataInterchange for MVS	3-6
DataInterchange for CICS	3-6
Receiving New Standards	3-7
DataInterchange for MVS	3-7
Considerations	3-7
DataInterchange for CICS	3-7
Chapter 4. Defining DataInterchange Operational Profiles	4-1
What is an Operational Profile?	4-1
Customizing National Language Support	4-2
Validation and Translation Tables	4-3
Single-Byte and Double-Byte Characters	4-3
Language Profile (LANGPROF)	4-3

User Program Information Profile (ADAMCTL)	4-5
MQSeries Queue Profile (MQSERIES)	4-7
Security Profile (SECUPROF)	4-9
Application Definition Profile (APPDEFS)	4-11
System Profile (SYSPROF)	4-14
Chapter 5. Establishing Communications with Trading Partners	5-1
Overview of Trading Partner Communications	5-1
Overview of the Minimal Trading Partners Feature	5-2
Setting Up the Network Profile	5-4
Connecting to GEIS	5-4
Connecting to IBM Global Network	5-5
Updating a Network Profile Member	5-5
Connecting to Other Networks	5-8
Adding a Network Profile Member	5-8
Using MQSeries Queues	5-13
Setting Up the Requestor Profile	5-14
Adding a Requestor Profile Member	5-14
Setting Up the Trading Partner Profile	5-23
Adding a Trading Partner Profile Member	5-23
Utilizing Control Numbers	5-39
Assigning Control Numbers by Standard Transaction ID	5-40
Group Control Number Masks	5-40
Trading Partner Determination	5-42
Handling a Trading Partner in Multiple Environments	5-42
Continuous Receive Facility	5-43
Setting Up the Continuous Receive Profile	5-44
Requesting, Sending, and Receiving Functional Acknowledgments	5-48
Requesting Functional Acknowledgments	5-48
Sending Functional Acknowledgments	5-48
Receiving Functional Acknowledgments	5-49
Delayed Functional Acknowledgments	5-50
Interchange Acknowledgments (TA1 Segment)	5-50
Chapter 6. Customizing EDI and Envelope Standards	6-1
Transactions/Messages, Segments, and Data Elements	6-1
Envelopes	6-3
Customizing EDI Standards	6-4
Copying an EDI Standard	6-7
Adding a Data Element or Composite Element	6-9
Updating a Composite Element	6-11
Adding a Segment	6-13
Adding a Transaction Set	6-15
Determining Loop IDs	6-18
Data Types for EDI Standard Data Elements	6-20
Adding an EDI Standard	6-22
Viewing Standards, Transaction Sets, Segments, Composite Elements, and Data Elements	6-23
Customizing Envelope Standards	6-26
Providing Envelope Data	6-26
Updating an Envelope Standard	6-27
Updating Envelope Segments	6-28
Using Data Types in Envelope Standards	6-29
Envelope Profiles	6-30

Adding a Member to an Envelope Profile	6-30
Data Types for Envelope Standards	6-32
EDIFACT Envelope Data	6-35
ICS Envelope Data	6-42
UNTDI Envelope Data	6-46
UCS Envelope Data	6-50
X12 Envelope Data	6-54
Building the Enveloping Segments	6-60
Chapter 7. Defining and Working With Your Application Data	7-1
Your Application Data	7-1
Record Formats Used by DataInterchange	7-1
Understanding Application Data Structures	7-3
DataInterchange Application Data Format	7-4
Application Data Format Worksheet	7-4
Adding an Application Data Format	7-7
Data Types for Application Data Formats	7-17
Reformatting Data into Control (C) and Data (D) Records	7-20
One D Record for Each Transaction	7-20
One D Record for Each Data Structure	7-21
Variable D Records	7-22
Changing Your Application Data Format after Mapping	7-23
Creating an Application Data Format for an Empty Transaction	7-23
Translating Application Data to Application Data (Fixed-to-Fixed)	7-23
Setting up Fixed-to-Fixed Translation	7-24
FIXED Standards without an Associated ADF	7-26
Blank Application Data Format Worksheet	7-26
Chapter 8. Translation and Validation Tables	8-1
Translation and Validation Tables - Usage and Tips	8-1
Tables Supplied by DataInterchange	8-2
ALPHANUM and CHARSET Tables	8-2
Alphanumeric Table (ALPHANUM)	8-3
Character Set Table (CHARSET)	8-3
File Name Table (FILENAME)	8-3
Monocase Table (MONOCASE)	8-3
Program Name Table (PRGMNAME)	8-3
Translation Tables	8-4
When to Use Type T Translation Tables	8-4
When to Use Type R (Reverse) Translation Tables	8-5
Adding a Translation Table	8-5
Updating a Translation Table	8-8
Validation Tables	8-9
Adding a Validation Table	8-10
Updating a Validation Table	8-12
Chapter 9. Mapping Your Application Data to an EDI Standard Transaction Set	9-1
What is Mapping?	9-1
What is a Trading Partner Transaction?	9-1
Overview of the Mapping Process	9-2
Before You Begin Mapping	9-2
EDI Segments and Data Elements	9-3
Using Accumulators	9-3

Using Literals	9-4
Using Literals for Send Mapping	9-5
Segment Creation for Send Mapping	9-5
Using Literals for Receive Mapping	9-6
Format of Literal Data	9-6
Accumulator Literals	9-6
Conditional Processing Literals	9-7
Literal Keywords	9-8
Named Variables	9-16
Expressions	9-17
Boolean Operators	9-18
Comparison Operators	9-18
Arithmetic Operators	9-18
Unary Operator	9-19
Special Operators	9-20
Date Conversion Special Operators	9-21
Order of Precedence	9-24
DI Variables	9-24
Mapping Techniques for Literal Keywords	9-26
Examples of Using Literal Keywords and Named Variables	9-27
Control Data Literals	9-33
Mapping Specific Service Segment Fields (Receive Only)	9-33
Mapping Generic Service Segment Fields (Receive Only)	9-34
Mapping Service Segment Fields (Send Only)	9-36
Validation During Mapping	9-37
Mapping Guidelines	9-38
Defining a Data Hierarchy for Mapping	9-38
Mapping Nonrepeating Segments	9-39
Mapping Repeating Segments Not in Loops	9-39
Mapping Loops	9-39
Single Occurrences	9-39
Multiple Occurrences	9-40
Repeating Segment Within a Loop	9-40
Mapping Data Elements to and from Structure Fields	9-40
Send Mapping One to Many, Many to One	9-41
Receive Mapping to Multiple Fields	9-41
Mapping Floating Segments	9-41
Processing Bounded Loops	9-42
Preparing the Mapping Worksheet	9-43
Creating a Trading Partner Transaction (Map)	9-45
Defining a Transaction	9-47
Mapping Segments and Data Elements	9-52
Specifying an Application Field Name	9-57
Specifying an Accumulator and Action	9-57
Specifying a Literal	9-58
Special Handling	9-59
Concatenating Data for a Data Element (Send Only)	9-61
Substringing Data for an Application Field (Receive Only)	9-64
Using Named Variables and Special Operators	9-65
Specifying Occurrences	9-67
Qualifying a Segment	9-68
Qualifying a Loop	9-69
Qualifying a Data Element	9-71

Specifying Trading Partners	9-78
Add Trading Partner Usage for Sending	9-78
Trading Partner Usage Overrides for Sending	9-82
Defining Generic Send Usages	9-85
Defining Send Usages	9-85
Determining the Trading Partner	9-86
Selecting the Send Usage	9-86
Standard Envelope Profile Member Selection	9-87
Transaction Store Considerations	9-88
Management Reporting Considerations	9-88
Export/Import Considerations	9-88
Add Trading Partner Usage for Receiving	9-88
Trading Partner Usage Overrides for Receiving	9-94
Defining Generic Receive Usages	9-96
Receive Usage Definition	9-96
Determining the Trading Partner	9-96
Selecting the Receive Usage	9-97
Functional Acknowledgment Processing	9-97
Transaction Store Considerations	9-97
Management Reporting Considerations	9-97
Export/Import Considerations	9-97
Migrating a Transaction Mapping	9-97
Migrating Trading Partner Usages	9-101
Generating Control Strings	9-103
Hierarchical Loops	9-105
The HL Segment	9-106
Preparing Hierarchical Loops	9-107
Literal Keywords for the HL Segment	9-109
Mapping the HL Segment	9-110
Hierarchical Loop Case - Ship Notice (856)	9-116
Hierarchical Loop Case - Ship Notice (856) with a Special Requirement	9-117
DataInterchange Inbound and Outbound Processing Flowcharts	9-118
 Chapter 10. Managing Your EDI Data Using the Transaction Store Facility	 10-1
Data Flow through the Transaction Store	10-2
Transaction Store Facility Menu (TF01)	10-6
Selecting EDI Transactions You Want to Work With	10-7
Default Criteria	10-8
Criteria Selections Panel (TF02)	10-8
Transaction Criteria	10-9
Trading Partner and Network Criteria	10-11
Envelope Criteria	10-12
Application Criteria	10-13
Store Status Criteria	10-14
Additional Selection Criteria Panel (TF08)	10-15
Sending and Receiving Transactions	10-15
Enveloping Transactions	10-15
Enveloping and Sending Transactions	10-16
Receiving Transactions	10-17
Translating Received Transactions	10-19
Managing Transactions	10-22
Reporting Transaction Status	10-23
Updating Store Status	10-24

Updating Network Status	10-25
Reporting Interchange or Group Status	10-25
Updating Interchange or Group Status	10-27
Transaction Status Codes	10-28
Panel Actions	10-31
Acknowledgment	10-31
Detail	10-31
Envelope	10-32
eVentlog	10-33
eXclude	10-33
grOups	10-33
Hold	10-33
Image	10-34
Line	10-34
Options	10-34
Print	10-35
purGe	10-36
Release	10-36
Report	10-36
Send	10-36
Summary	10-37
Transactions	10-37
Translate	10-37
Unpurge	10-37
Chapter 11. Exporting and Importing Transactions	11-1
Overview of the Export and Import Process	11-1
Export and Import File List	11-3
Export and Import Restrictions and Limitations	11-3
Using the Export Menu (EI00) Options	11-4
Exporting Standards	11-5
Exporting Application Data Formats	11-7
Exporting Trading Partner Transactions (Maps)	11-7
Exporting Control Strings	11-10
Exporting Business Document Definitions	11-12
Exporting Business Document Layouts	11-14
Exporting Profiles	11-14
Exporting Tables	11-16
Using the Import Menu (EI01) Options	11-16
Importing Standards	11-17
Importing Application Data Formats	11-18
Importing Trading Partner Transactions (Maps)	11-19
Importing Control Strings	11-20
Importing Business Document Definitions	11-21
Importing Business Document Layouts	11-22
Importing Profiles	11-22
Importing Tables	11-23
Printed Report	11-24
Export/Import Facility and Utility Comparisons	11-25
Chapter 12. Event Logging	12-1
Understanding Log Profiles	12-1
Activity Log Profile	12-1

Printing or Viewing Log Entries	12-4
Interpreting Log Entries	12-7
Translator Messages	12-7
Turning Logging On and Off	12-8
Defining an Event Log	12-9
 Chapter 13. Using the Interactive Entry Facility (IEF)	13-1
Before You Can Use IEF	13-3
Interactive Entry Facility Menu (MM01)	13-3
Working with Business Document Definitions	13-4
Adding a Business Document Definition	13-4
Updating, Viewing, or Deleting a Business Document Definition	13-6
Working with Business Documents	13-9
Business Documents Panel (DE01)	13-9
Titles of Business Document Panels	13-10
Business Document Status	13-10
Business Document Functions and Status	13-12
Navigating within a Business Document	13-13
Adding a Business Document	13-13
Sending a Business Document	13-17
Copying a Business Document	13-18
Deleting a Business Document	13-18
Filing, Archiving, or Getting (Restoring) a Business Document	13-19
Printing a Business Document	13-22
Updating a Business Document	13-22
Viewing a Business Document	13-23
Recalling a Business Document from the Network	13-24
Receiving a Business Document	13-25
Translating a Business Document to EDI Format	13-26
Working with Messages	13-26
Messages Panel (DM01)	13-26
Message Status	13-27
Message Functions and Status	13-28
Adding a Message	13-28
Sending a Message	13-29
Copying or Updating a Message	13-30
Deleting a Message	13-31
Printing a Message	13-31
Viewing a Message	13-31
Recalling a Message	13-31
Receiving a Message	13-32
Updating the Status of Messages	13-32
Handling File Requests	13-33
File Requests Panel (DF01)	13-33
File Request Status	13-34
File Request Functions and Status	13-34
Adding a Request to Send a File	13-35
Sending a File	13-36
Deleting a File Request	13-37
Viewing a File Request	13-37
Recalling a File	13-38
Receiving a File	13-38
Updating the Status of File Requests	13-39

Chapter 14. Customizing Business Documents	14-1
Starting the Document Layout Utility	14-1
Input to the Document Layout Utility	14-2
The Application Data Format	14-2
The Layout Statements	14-2
Syntax Rules	14-2
LAYOUT	14-3
GROUP	14-3
ENDGROUP	14-4
STRUCT	14-5
LITERAL	14-5
FIELD	14-5
Output from the Document Layout Utility	14-7
Printing Customized Documents in Batch	14-7
Document Layout Example	14-7
The Sample Application Data Format	14-7
The Sample Purchase Order	14-8
Including IEF Control Fields in the Layout	14-9
Suppressing Individual Fields	14-9
Defining Fields that Require More than One Line	14-10
Suppressing Groups of Fields and Literals	14-11
Defining Repeating Structures	14-12
Printing Page Headings	14-13
The Complete Layout Statement File	14-14
Predefined Layouts	14-16
The Purchase Order Layout	14-16
The Invoice Layout	14-18
Document Layout Messages	14-20
 Chapter 15. Invoking the DataInterchange Utility	15-1
Invoking the Utility from the Administrator's Menu	15-2
 Appendix A. Security	A-1
RACF Class for DataInterchange	A-1
The Resource Names	A-1
Defining Levels of Access	A-2
Controlling Access to Menu Options	A-2
Administrator's Menu (MP01)	A-3
Interactive Entry Facility Menu (MM01)	A-4
Transaction Store Facility Menu (TF01)	A-4
Profile Data	A-5
Event Logging	A-5
Trading Partner Transactions	A-6
EDI Standards	A-6
Application Data Formats	A-6
Translation and Validation Tables	A-6
Envelope Standards	A-7
Translation and Communication Services	A-7
Export/Import Utility	A-8
Migrating a Transaction Mapping	A-8
Archiving/Restoring Event Log Records and Removing Store Documents	A-8
Where Resource Names Are Checked	A-8
Security Examples	A-11

Using RACF General Resources Services	A-13
Appendix B. Using the 841 Transaction Set	B-1
The BIN Segment ID	B-1
Length of the BIN Segment	B-1
The EFI Segment	B-1
Send Processing for the Binary Segment	B-2
Mapping Data from a File to a Binary Segment	B-2
Format Specifications	B-3
Examples	B-5
Sending PC Executable Files	B-5
Sending Variable Length Files to PCs	B-5
Sending a File to a System with a Limit	B-5
Mapping an Application Field to a Binary Segment	B-5
Receive Processing for the Binary Segment	B-7
Mapping Data from a Binary Segment to a File	B-7
Mapping Data from a Binary Segment to an Application Field	B-9
Appendix C. The IBM DataInterchange Web Site	C-1
Glossary	X-1
Index	X-7

Figures

1-1.	Extending Your Enterprise	1-2
1-2.	DataInterchange Services Overview	1-4
3-1.	Obtaining Standards Electronically Flowchart	3-2
6-1.	Paper Format and EDI Standard Format of a Transaction	6-3
6-2.	Levels of Enveloping	6-4
6-3.	Data Sources for Envelopes	6-26
6-4.	Building the ISA segment (Override)	6-61
6-5.	Building the ISA Segment (Default)	6-62
6-6.	Building the GS Segment	6-63
6-7.	Building the ST Segment	6-64
6-8.	Building the UNB Segment (Override)	6-65
6-9.	Building the UNB Segment (Default)	6-66
6-10.	Building the UNG Segment	6-67
6-11.	Building the UNH Segment	6-68
7-1.	Sample Application Data with Record Identifier	7-1
7-2.	Sample Application Data with C and D Records	7-2
7-3.	Reformatted Data	7-2
7-4.	Sample Application Data	7-3
7-5.	Sample Working Storage Record Definition of a Purchase Order	7-6
7-6.	Data Format for the Sample Purchase Order	7-16
7-7.	Reformatted Data—All Structures Passed Together	7-20
7-8.	Reformatted Data—All Structures Passed Separately	7-21
7-9.	Reformatted Data—Some Structures Passed Separately	7-22
7-10.	Fixed-to-Fixed	7-26
9-1.	Steps in the Mapping Process	9-2
9-2.	Required Data Elements	9-3
9-3.	Sample Purchase Order in EDI Format	9-43
9-4.	Record Layout for Sample Purchase Order (Sending)	9-44
9-5.	Mapping Worksheet for Sample Purchase Order (Receiving)	9-44
9-6.	Sample Purchase Order Application Data Format (Send)	9-46
9-7.	Sample Purchase Order Application Data Format (Receive)	9-47
9-8.	Receive Transaction Mapping Printout	9-104
10-1.	Data Flow through Transaction Store for Sending	10-3
10-2.	Data Flow through Transaction Store for Receiving	10-5
10-3.	Status Summary Report	10-24
11-1.	The Export/Import Process	11-2
11-2.	Sample Report - DataInterchange Export	11-24
11-3.	Sample Report - DataInterchange Import	11-25
12-1.	Sample Translator Message	12-7
12-2.	Sample Activity Log Profile Member	12-9
13-1.	IEF Components	13-2
14-1.	Purchase Order Layout	14-17
14-2.	Invoice Layout	14-19

Tables

1-1.	Values for Type Field	1-14
1-2.	DataInterchange Terminology Changes	1-15
5-1.	Network Profile Members	5-4
5-2.	Special Codes for Group Control Number Masks	5-40
6-1.	Summary of Tasks for EDI and Envelope Standards	6-5
6-2.	Tasks for Modifying Transaction Sets, Segments, and Elements	6-7
6-3.	Data Types for EDI and Envelope Standards	6-21
6-4.	Data Types for Envelope Standards Only	6-33
6-5.	EDIFACT Envelope Data	6-35
6-6.	ICS Envelope Data	6-42
6-7.	UNTDI Envelope Data	6-46
6-8.	UCS Envelope Data	6-50
6-9.	X12 Envelope Data	6-54
7-1.	Application Data Worksheet for Sample Purchase Order	7-6
7-2.	Data Types Used in Data Formats	7-17
7-3.	Blank Worksheet for Your Application Data Format	7-27
9-1.	Literal Keywords	9-8
9-2.	Service Segments	9-33
9-3.	Keywords for Mapping Envelope Data (Receive Only)	9-34
9-4.	Type of Validation During Translate to Standard	9-37
9-5.	Type of Validation During Translate to Application	9-38
9-6.	Steps for Creating a Trading Partner Transaction (Map)	9-45
9-7.	HL Segment	9-107
10-1.	Transaction Store Facility Menu (TF01) Options	10-6
10-2.	Default Criteria for Transaction Store Facility Menu (TF01) Options	10-8
10-3.	Transaction Status Codes	10-28
10-4.	Valid Store Services for Status Codes	10-29
13-1.	Interactive Entry Facility Menu (MM01) Options	13-4
13-2.	Business Document Status	13-10
13-3.	Business Document Functions and Status	13-12
13-4.	Message Status	13-27
13-5.	Message Functions and Status	13-28
13-6.	File Request Status	13-34
13-7.	File Request Functions and Status	13-35
A-1.	Where Resource Names are Checked	A-8

Notices

This publication was developed for products and services offered in the United States. IBM may not offer the products, services, or features discussed in this document in other countries. Consult your local IBM representative for information on the products and services currently available in your area. Any reference to an IBM product, program, or service is not intended to state or imply that only an IBM product, program, or service may be used. Any functionally equivalent product, program, or service that does not infringe on any of IBM Global Services' intellectual property rights may be used instead. However, it is the user's responsibility to evaluate and verify the operation of any non-IBM product, program, or service.

IBM may have patents or pending patent applications covering subject matter described in this document. The furnishing of this document does not give you any license to these patents. You can send license inquiries, in writing to:

IBM Director of Licensing
IBM Corporation
500 Columbus Avenue
Thornwood, NY 10594
U.S.A.

The following paragraph does not apply to the United Kingdom or any other country where such provisions are inconsistent with local law: INTERNATIONAL BUSINESS MACHINES CORPORATION PROVIDES THIS PUBLICATION "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Some states do not allow disclaimer of express or implied warranties in certain transactions, therefore, this statement may not apply to you.

This publication could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes will be incorporated in new editions of the publication. IBM may make improvements and/or changes in the product(s) and/or the program(s) described in this publication at any time without notice.

Trademarks and Service Marks

The following terms are trademarks or service marks of the International Business Machines Corporation in the United States and other countries:

Expedite
IBM
IBM Global Network
IBM Global Services

All other trademarks and service marks are the property of their respective owners.

Softcopy Agreement

For softcopy versions of this book, we authorize you to:

- Copy, modify, and print the documentation contained on the media, for use within your enterprise, provided you reproduce the copyright notice, all warning statements, and other required statements on each copy or partial copy.

- Transfer the original unaltered copy of the documentation when you transfer the related IBM product (which may be either machines you own, or programs, if the program's license terms permit a transfer). You must, at the same time, destroy all other copies of the documentation.

You are responsible for payment of any taxes, including personal property taxes, resulting from this authorization.

THERE ARE NO WARRANTIES, EXPRESS OR IMPLIED, INCLUDING THE WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE.

IN NO EVENT WILL IBM BE RESPONSIBLE FOR ANY CONSEQUENTIAL OR INCIDENTAL DAMAGES ASSOCIATED WITH USE OF THE PROGRAM OR DOCUMENTATION, EVEN IF IBM HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

Some jurisdictions do not allow the exclusion of implied warranties, so the above exclusion may not apply to you.

Your failure to comply with the terms above terminates this authorization. Upon termination, you must destroy your machine readable documentation.

To the Reader

This book describes the administrative tasks for DataInterchange for MVS and DataInterchange for CICS Version 3 Release 1. It provides information for the administrator on how to define the environment, customize standards, define trading partners and application data, and map transactions or messages in DataInterchange. This book also describes how to use other DataInterchange features to import data to your system, export data from your system, customize business documents to meet your specific needs, and invoke the DataInterchange Utility.

IBM provides IBM Global Network Services only in the United States. In other countries, another company may provide IBM Global Network Services. References in this document to your marketing representative refer to your IBM Global Network representative.

Changes after Publication

The README file on the product tape may contain additional information or changes made after this book was published.

Who Should Read This Book

This book is intended for the electronic data interchange (EDI) administrator who prepares a computer system for electronic exchange of business information. The administrator should be familiar with, or have a working knowledge of:

- EDI standards
- Translation of application data to an EDI standard format
- Application programs that interface with DataInterchange
- Agreements with your trading partners
- Networks used for exchanging information

Related Books

The following books contain information related to the topics covered in this guide:

- *Customizing and Developing Applications with Expedite/CICS*, GC34-3304
- *DataInterchange Client User's Guide*, SB34-2010
- *DataInterchange Installation Guide*, GB09-8070
- *DataInterchange Messages and Codes*, SB34-2000
- *DataInterchange Programmer's Reference*, SB34-2001
- *Expedite Base/MVS Programming Guide*, GC34-2204
- *Licensed Program Specifications (CICS)*, GB35-4045
- *Licensed Program Specifications (MVS)*, GB09-8079
- *Understanding Information Exchange Message Charges*, GX66-0653
- *Using Information Exchange Administration Services*, GC34-2221

What EDI Courses are Available

This book is designed for readers who have an understanding of EDI and communication fundamentals, standards, and data mapping. IBM offers the following Electronic Data Interchange (EDI) courses in the United States to obtain these prerequisites.

EDI Concepts, Standards, and Mapping (course M112)

This course introduces you to Electronic Commerce and the role of Electronic Data Interchange (EDI). Students gain a working knowledge of EDI standards, standards organizations, standards terminology, and mapping processes in preparation for implementing or assessing EDI translation software. No specific software is taught in this class. However, the function of and requirements for EDI software are explored in detail. The course addresses all major EDI standards, national and international, with emphasis on the ASC X12 standard. Learning is reinforced through classroom exercises using EDI standards documentation.

You will learn how to:

- Define Electronic Commerce and the EDI process
- List the benefits of implementing EDI and view industry examples of its use
- Define the functional components of an EDI system
- Identify major standards organizations and their structure, functions, and procedures
- Use EDI standards definitions to define a trading partner transaction (mapping)
- Identify the functions of EDI standard control structures
- Map an entire document to EDI segments and elements
- Describe the security processes available for EDI
- Identify the various alternatives available for integrating the Internet into your Electronic Commerce Strategy

DataInterchange for MVS Implementation (course M113)

This course will provide you with the skills you need to set up and implement DataInterchange/MVS after installation. This is accomplished via classroom lecture and hands-on machine exercises. The IBM Global Network will be used as the model network in the lab exercises, although primary emphasis is on the DataInterchange product.

Students are also introduced to the DataInterchange Client Graphical User Interface (GUI) by way of product demonstration. After each lab in the DataInterchange mainframe environment, students are shown how the same function is performed in the DataInterchange Client environment.

You will learn how to:

- Define the terms and concepts used in the EDI process
- Perform the necessary setup and customization to successfully map and translate application data to EDI-formatted data using DataInterchange for MVS and DataInterchange for CICS
- Tailor the JCL for Expedite Base/MVS and code the necessary commands to communicate with a specified trading partner
- Set up and describe the flow of translation and communication for an environment in which DataInterchange/MVS and Expedite Base/MVS are both installed for use together

- List the implementation differences between DataInterchange for MVS and DataInterchange for CICS
- Describe the features and functions of the DataInterchange Client environment
- Compare and contrast the DataInterchange mainframe interface to the DataInterchange GUI interface

Contact an IBM marketing representative for schedules and fees, or call IBM education at 1-800-880-2335.

Similar courses are available outside the United States. For information on their availability and content, contact a representative of the IBM Global Network in your country.

How This Book Is Organized

This book contains the following information:

Chapter 1, “Introducing DataInterchange,” introduces the concepts and working relationship of EDI, DataInterchange, and DataInterchange Client.

Chapter 2, “Getting Started with DataInterchange,” describes how to access DataInterchange for MVS or DataInterchange for CICS, and describes the DataInterchange panels.

Chapter 3, “Importing and Requesting EDI Standards Electronically,” provides information about adding standards to your database and requesting standards electronically from the IBM Global Network.

Chapter 4, “Defining DataInterchange Operational Profiles,” describes how to define the operational profiles, such as language, user program information, and trading partner authorizations.

Chapter 5, “Establishing Communications with Trading Partners,” describes how to define information about your network, yourself, and your trading partners to DataInterchange. This chapter also discusses the Continuous Receive Facility available in CICS.

Chapter 6, “Customizing EDI and Envelope Standards,” describes EDI and envelope standards and provides information about customizing standards.

Chapter 7, “Defining and Working With Your Application Data,” provides information about data formats and describes how to analyze application data.

Chapter 8, “Translation and Validation Tables,” provides information about translation and validation tables, and describes how to create them.

Chapter 9, “Mapping Your Application Data to an EDI Standard Transaction Set,” provides information about trading partner transactions and messages, and describes how to map transactions or messages.

Chapter 10, “Managing Your EDI Data Using the Transaction Store Facility,” provides information about the Transaction Store, and describes how to use the Transaction Store Facility to send, receive, and manage transactions or messages in the Transaction Store.

Chapter 11, “Exporting and Importing Transactions,” provides information about the export and import functions and describes the export and import options.

Chapter 12, “Event Logging,” provides information about logging and archiving events.

- | Chapter 13, "Using the Interactive Entry Facility (IEF)," describes how to use the Interactive Entry Facility to create business documents and messages, and how to send and receive them.
- | Chapter 14, "Customizing Business Documents," describes how to use the Document Layout facility to customize business document layouts.
- | Chapter 15, "Invoking the DataInterchange Utility," describes how to invoke the DataInterchange Utility from the Administrator's menu or by using a CLIST.
- | Appendix A, "Security," describes how to protect DataInterchange resources using the Resource Access Control Facility (RACF).
- | Appendix B, "Using the 841 Transaction Set," provides information about using the ASC-X12 Specifications/Technical Information transaction set, 841, to allow trading partners to exchange technical information.
- | Appendix C, "The IBM DataInterchange Web Site," provides information about the IBM DataInterchange web site and its features.

Where to Find More Information

- | For a list of related publications, see "Related Books" on page xix.

You may also refer to the DataInterchange Client product manuals.

Information about Standards

Using EDI requires detailed information about transaction and message standards that are not documented in this book. Sources of information about EDI standards include:

Standard	Address
X12	American National Standards Institute New York, New York 10018 Sales Department (212) 642-4900
EDIFACT	International Organization for Standardization 1 rue de Varembe Case Postale 56 CH-1121 Geneva 20, Switzerland
ODETTE	Organization for Data Exchange by Teletransmission in Europe Forbes House, Halkin Street London SW1X7DS
UN/EDI (TRADACOMS)	UN/EDI standards organization address TRADACOMS standards organization address Article Number Association (UK) Ltd. 11 Kingsway London WC2B 6AR 71-240-2974
EDIA (TDCC)	The Electronic Data Interchange Association Suite 550 225 Reinekers Lane Alexandria, VA 22314 703-838-8042

Standard	Address
UCS and WINS	Uniform Communication Standard Warehouse IBM Global Network Standard P.O. Box 1244 Dayton, OH 45401

Terms Used in This Book

The following abbreviations and terms are used in this book:

Abbreviation/Term	Definition
Account number	Account identifier
Applications	Programs that process information
CICS	Customer Information Control System
Client/Server	The model of interaction in distributed data processing in which a program at one site sends a request to a program at another site and awaits a response. The requesting program is called a client; the answering program is called a server.
Data set	The basic unit of data storage for MVS
DataInterchange	DataInterchange for MVS and DataInterchange for CICS
DI Client	DataInterchange Client interface, a Windows-based graphical user interface for DataInterchange
DI Host	DataInterchange for MVS or CICS mainframe products
DI for MVS	DataInterchange for MVS
DI for CICS	DataInterchange for CICS
EDI	Electronic data interchange
Electronic transmission	A method of transmitting data, such as a public network
Facility	DataInterchange processes and programs that are interactive
Host	The mainframe environment (MVS or CICS)
MVS	Multiple Virtual Storage operating system
Profile	A collection of descriptive information
TSO/E	Time Sharing Option/Extended
Utility	DataInterchange processes and programs that run in batch mode

ANSI X12 and EDIFACT Terminology

In all matters relating to EDI standards, this book tries to match ANSI X12 terminology. In a few cases, EDIFACT uses different terms or terms that do not have X12 equivalents. Some of these differences are:

X12 Term	EDIFACT Term	DataInterchange Version 3 Release 1 Term
Communication session	Connection	Communication session
Transaction set	Message	Transaction or message
Segment ID	Tag	Segment ID
Loop	Group	Loop

X12 Term	EDIFACT Term	DataInterchange Version 3 Release 1 Term
Control segment	Service segment	Envelope

The glossary at the end of this book defines additional terms associated with DataInterchange.

Syntax Conventions Used in This Book

The following syntax conventions are used throughout this book.

- In text, uppercase letters in bold represent values that you type without change. These values are not case-sensitive. For example:

EDI PRINT(FILE)

Indicate the data type: **CH** or **R**.

Specify the literal **&SAVE QTY1**.

- On panels, reverse block letters represent values that you type. For example:

userid

- Lowercase italicized letters represent variable parameters for which you supply the values. For example:

SYSID(*system-name*)

- Italicized letters are also used for emphasis within textual descriptions. For example:

EDI standards define the segments and data elements...

Understanding Syntax Diagrams

This section describes how to read the syntax diagrams in this book.

Getting Started: To read a syntax diagram, follow the path of the line. Read from left to right and top to bottom.

- The ►— symbol indicates the beginning of a syntax diagram.
- The —► symbol, at the end of a line, indicates that the syntax diagram continues on the next line.
- The ►— symbol, at the beginning of a line, indicates that a syntax diagram continues from the previous line.
- The | symbol indicates the beginning and end of a fragment, or part of the command syntax.
- The —►◀ symbol indicates the end of a syntax diagram.

Syntax items (for example, a keyword or variable) may be:

- Directly on the line (required)
- Above the line (default)
- Below the line (optional)

Syntax Diagram Description

Example

Abbreviations:

Uppercase letters denote the shortest acceptable abbreviation. If an item appears entirely in uppercase letters, it cannot be abbreviated.

You can type the item in uppercase letters, lowercase letters, or any combination.

In this example, you can enter KEYWO, KEYWOR, or KEYWORD in any combination of uppercase and lowercase letters.



Symbols:

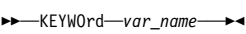
You must code these symbols exactly as they appear in the syntax diagram.

- * Asterisk
- :
- ,
- = Equal Sign
- Hyphen
- () Parentheses
- .

Variables:

Highlighted lowercase items (*like this*) denote variables.

In this example, *var_name* represents a variable you must specify when you code the KEYWORD command.

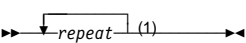
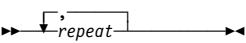
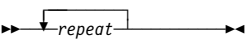


Repetition:

An arrow returning to the left means that the item can be repeated.

A character within the arrow means you must separate repeated items with that character.

A footnote (1) by the arrow references a limit that tells how many times the item can be repeated.



Note:
1 Specify *repeat* up to 5 times.

Required Choices:

When two or more items are in a stack and one of them is on the line, you *must* specify one item.

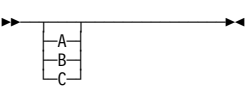
In this example, you must choose A, B, or C.



Optional Choice:

When an item is below the line, the item is optional. In this example, you can choose A or nothing at all.

When two or more items are in a stack below the line, all of them are optional. In this example, you can choose A, B, C, or nothing at all.



Defaults:

Defaults are above the line. The system uses the default unless you override it. You can override the default by coding an option from the stack below the line.

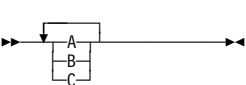
In this example, A is the default. You can override A by choosing B or C.



Repeatable Choices:

A stack of items followed by an arrow returning to the left means that you can select more than one item or, in some cases, repeat a single item.

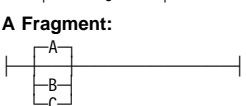
In this example, you can choose any combination of A, B, or C.



Syntax Fragments:

Some diagrams, because of their length, must fragment the syntax. The fragment name appears between vertical bars in the diagram. The expanded fragment appears in the diagram after a heading with the same fragment name.

In this example, the fragment is named "A Fragment."



Sending Comments for This Publication

You can send comments using the form in the back of this publication, or use electronic mail or fax. Instructions are included for submitting your comments by mail, fax, or electronically.

Summary of Changes

This edition of the *DataInterchange Administrator's Guide* is based on the DataInterchange for MVS and DataInterchange for CICS product set. The changes that were made since the last edition was produced are identified within the text by a vertical bar (|).

| Summary of Changes for DataInterchange Version 3 Release 1

| This list summarizes the changes made to DataInterchange in Version 3 Release 1.

- | • Provided additional support for the Year 2000.
- | • Implemented MQSeries for application files.
- | • Added SAPUPDT field in Continuous Receive profile.
- | • Provided full SAP integration.
- | • Changed existing R data type in X12 for exponential notation.
- | • Provided for the Carbon Copy values (I) in ISA15 field.
- | • Implemented DT element length change, now 6 or 8.
- | • Provided delivery priority to requester and trading partner profiles, primarily for Expedite Base and Expedite CICS.
- | • Added minimal trading partners.
- | • Provided additional trading partner profile data.
- | • Implemented full standards compliance checking.
- | • Added document ID.
- | • Added pageable translation.
- | • Provided control numbers by trading partner pairing.
- | • Added integrated MQSeries support.
- | • Changed duplicate account user ID message to name trading partner.
- | • Allowed rename of trading partner user fields 1 through 10.
- | • Provided for multiple ALPHANUM/CHARSET tables.
- | • Allowed client access to envelope profiles.
- | • Allowed client access to envelope standards.
- | • Added fixed-to-fixed mapping support to client.
- | • Added event log viewing to client.
- | • Showed mandatory segments and elements on mapping details of client.
- | • Increased trading partner requestor retention period value from 2 to 3 characters.
- | • Added string search capability within transaction store images.

Summary of Changes for DataInterchange Version 2 Release 1

This list summarizes the changes made to DataInterchange in Version 2 Release 1.

Host Enhancements:

- Interface to the SDM LinkPlus file delivery system

- User controlled condition codes

This enhancement allows you to override DataInterchange generated condition codes with new values.

- Envelope data switch - send functional acknowledgment back to originator of the document

This enhancement allows you to tell DataInterchange to move inbound envelope data to the outbound functional acknowledgment. This may eliminate the need for FAENV and reduce the number of envelope profiles required.

- Generic trading partner usages

This enhancement allows you to define a single usage that can be used for multiple trading partners.

- Availability of profiles and tables in DB2 format as well as VSAM

- Import of standards by transaction

This function was previously provided in the Standards Apply facility, which was removed as of DataInterchange Version 3 Release 1.

- Expedite Comm-Press support for data compression

DataInterchange now provides a flag that is passed to Expedite Base/MVS to allow data compression using the Comm-Press data compression software.

- Composite Data Element support

This enhancement allows you to define and update composite data elements throughout a standard.

- Additional flags and warnings on the Transaction Mapping panel (TP01), the Export TP Transactions panel (EI06), and the Export Control Strings panel (EI10). This enhancement helps differentiate between maps and control strings maintained on the host and those maintained using the DataInterchange Client interface.

- Ability to import profiles by member

- Ability to import files that have associated usages or profiles

DataInterchange Client has an "Export with associated members" capability. DataInterchange host can import such a file.

- Ability to ignore control numbers in an import file when importing trading partner profiles

With previous releases of DataInterchange, existing control numbers were overwritten with those in the import file.

- Support for more than 99 standard data elements

- Classification of user exits

- Availability of DataInterchange Client, a Windows-based, graphical user interface for DataInterchange.

See "Using DataInterchange with the DataInterchange Client Graphical User Interface" and the DataInterchange Client documentation for information about this new interface.

Changes to the DataInterchange Host

Standards Apply

- | With the release of DataInterchange for MVS Version 3 Release 1 and DataInterchange for CICS Version 3 Release 1, the Standards Apply facility is no longer used to load standards; instead, the Import facility is used. Enhancements to Import allow you to select individual transactions to import without having to import the entire standard.
- | You must use the CLIST parameter STDADM(y) to prepare for the import of standards. See Chapter 3, "Importing and Requesting EDI Standards Electronically," for more information.

Export/Import Fixed Record Formats

The Export/Import fixed record formats have been changed. All fields defined with type HEX and length 1 have been changed to type CHAR and length 2. Additional fields have been added to the end of some record formats. Additional record formats have been added. For more information, see the Export/Import section in *DataInterchange Programmer's Reference*.

Management Reporting Data Extracts

Trading Partner Capability Data Extract, Network Activity Data Extract, and Transaction Activity Data Extract record layouts changed to expand total transactions and total errors fields. Lengths changed from 11 to 15.

Export/Import Changes

Several changes were made to DataInterchange, which affect export/import processing. Be aware of the following:

- The required size of the export/import file has changed from RECFM=VB, LRECL=4089, BLKSIZE=4093 to RECFM=VB, LRECL=8152, BLKSIZE=8156
- Fixed format export records changed to show hexadecimal values in two characters instead of one character.
- DataInterchange Version 2 Release 1 control strings are exported in TAGGED format only. DI Version 2 Release 2 can still import a Version 1 Release 4 or Version 1 Release 5 control string in native form.

Important Notice to DataInterchange API Users

Because of changes in the DataInterchange code, several changes were made to the DataInterchange copybooks. As a result, you will need to recompile your API programs. In addition, you will need to make changes to your API programs if you wish to take advantage of the new features. Please review the following copybook changes:

- Translator control block - copybook EDITRCB - blkname TRCB
 1. Additional interchange envelope override fields have been added to the translator control block for outbound translator processes. These fields were added to the areas of the control block defining current interchange override fields.
 2. The CICS suspend control fields were also defined in the translator control block along with a flag to clear the errcdes array returned to the application.

3. Existing field names were not changed. Reserved area field names were changed.
 4. The length of the control block (1536) was not changed.
- Trading Partner Data Block - copybook EDITPDB - blkname TPPDB
 1. Reserved area at the end of the control block was not documented previously. An incorrect length causes translator services to use the block length for DataInterchange Version 1 Release 4.
 2. The length of the control block should be 1532.
 - Network Data Block - copybook EDINPDB - blkname NPDB
 1. Reserved area at the end of the control block was not documented previously.
 2. The length of the control block should be 300.
 - Requestor Data Block - copybook EDIRQDB - blkname REQDB
 1. Reserved area at the end of the control block was not documented previously.
 2. The length of the control block should be 264.

Note: Before DataInterchange Version 2 Release 1, reserved areas at the end of copybooks were excluded from the samples provided. With DataInterchange Version 2 Release 2, these reserved areas were added to the documentation and shipped copybooks. These reserved areas should be added to your copybooks to minimize future impacts of API changes.

Documentation for these enhancements is in the following books:

- | • *DataInterchange Administrator's Guide*
- | • *DataInterchange Programmer's Reference*
- | • *DataInterchange Installation Guide*
- | • *DataInterchange Messages and Codes*

Chapter 1. Introducing DataInterchange

Doing Business Through EDI	1-1
DataInterchange: An EDI Solution	1-2
Before You Begin	1-5
Setup for Sending and Receiving Data	1-5
Sending Your Data to a Trading Partner	1-6
Translating Application Data to Standard	1-6
Collecting Application Data for Translation	1-6
Translating Application Data as it is Generated	1-7
Enveloping and Sending Transactions	1-7
Envelope Overrides	1-7
Receiving Data From Your Trading Partner	1-8
Receiving and Deenveloping Transactions	1-8
Translating Standard Data to Application Format	1-9
Continuous Receive	1-9
Translating Application Data to Application Data	1-9
Functional Acknowledgments	1-9
Update Status	1-10
DataInterchange Reporting on Your EDI System	1-10
Online Inquiries Using the Transaction Store Facility	1-10
Formatted Reports Using the DataInterchange Utility	1-10
Transaction and Envelope Data Extracts Using the DataInterchange Utility	1-11
Management Reporting Data Extracts	1-11
Migrating Maps	1-12
Export/Import	1-12
Interactive Entry Facility (IEF)	1-12
Using DataInterchange with the DataInterchange Client Graphical User Interface	1-13
Client/Server Mode versus Standalone Mode	1-14
DataInterchange Client Databases	1-15
Naming Convention Changes for DataInterchange Client	1-15
Unavailable Functions in the DataInterchange Client Interface	1-15
Moving to DataInterchange Client	1-16
Moving Setup Profiles and Trading Partner Profiles	1-16
Moving Standards, Data Formats, and Maps	1-16

Chapter 1. Introducing DataInterchange

The accurate and timely flow of information in today's business environment presents a monumental challenge. As business processes evolve, your company's ongoing success depends on the flow of information. To improve these processes, you can exchange information electronically from one computer system to another. *Electronic data interchange (EDI)* is the bridge that makes this possible.

Doing Business Through EDI

EDI streamlines your business and gives you a competitive advantage in the marketplace by providing:

- Improved ability to target new opportunities
- Automated closed-loop processes with end-to-end application integration
- Reduced costs through:
 - Increased productivity
 - Flexibility in exchanging information with your business partners
 - Just-in-time inventory management
- Greater control throughout the business cycle

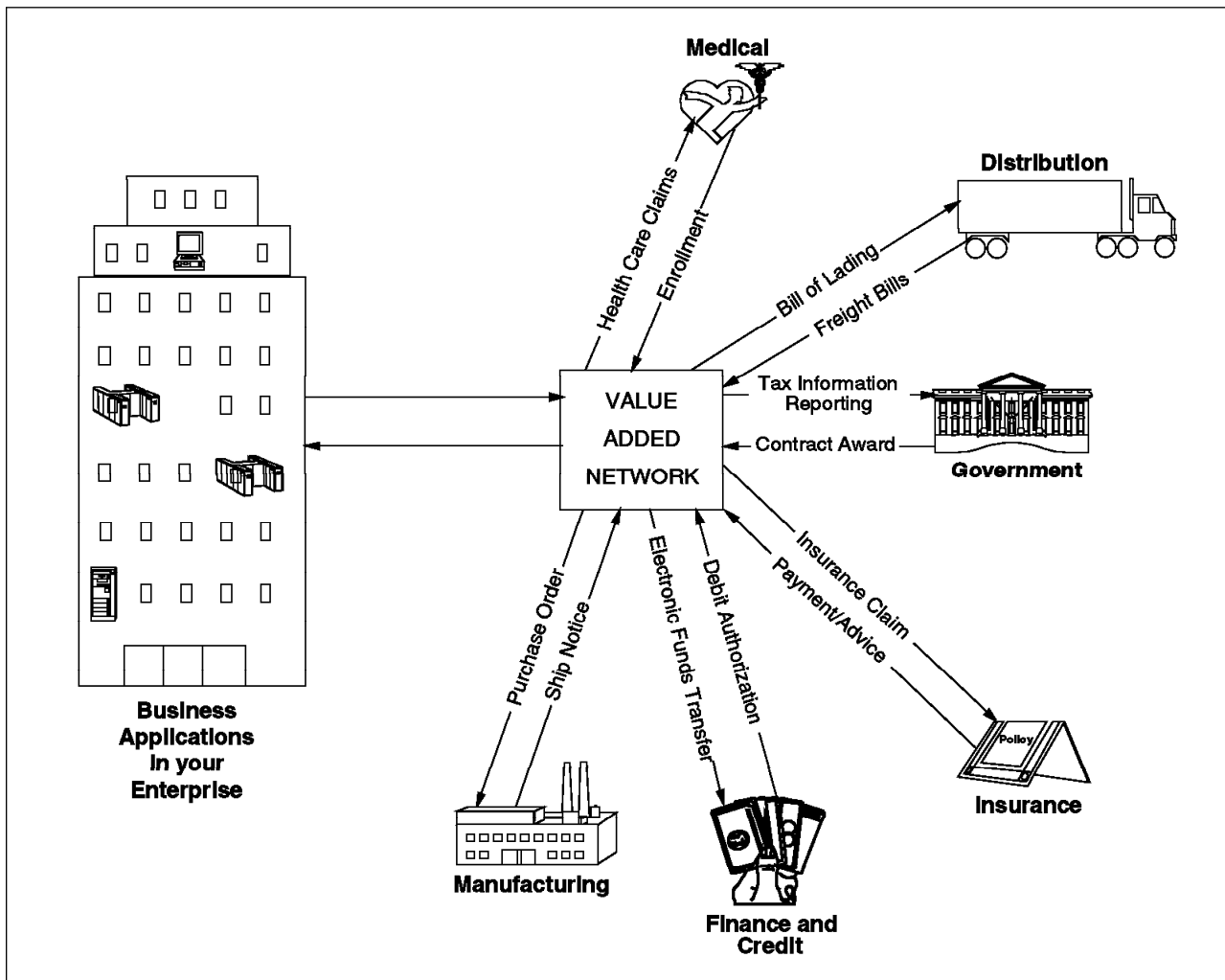


Figure 1-1. Extending Your Enterprise

DataInterchange: An EDI Solution

DataInterchange is a key part of your EDI implementation to automate your business processes and extend your enterprise directly to your trading partners. DataInterchange is a single application that reformats data for electronic transmission, and includes the following features:

Flexible Setup and Administration

- Online customization of standards, maps, and trading partner relationships
- Mapping designed to support:
 - Literals/constants
 - Accumulators, arithmetic and logical operations
 - Qualified loop and element mapping
 - Hierarchical loop mapping
 - Envelope field mapping
 - User exits at the field level
 - User-defined translation and validation tables
 - Boolean logic

- Maps can be used by one or more trading partners
- Export/import available to move user data between test and production systems

Superior Translation Capability

- Syntax checking
- Test and production support
- Ability to translate and envelope separately
- Flexible command language interface
- Interactive, batch, event-driven, and real-time processing
- Automatic generation of functional acknowledgments

Versatile Communications

- Support for networks including IBM Global Network, GEIS, and direct connections to trading partners
- Ability to resend individual transactions or entire envelopes
- Support for MQSeries Queues as a means of exchanging data between trading partners

Extensive Reporting and Auditing

- Reporting of trading partner relationships, including what transaction sets are being used, and when was the last communication with a trading partner, and others
- Reporting of envelope and transaction status for both online and batch processing
- Exception reporting
- Setting acceptable error levels for the trading partner/map combination
- Reporting of SAP status for online and batch processing
- Optional audit log with archive recovery capability

Standards Support

- Multiple standards, including EDIFACT, X12, ODETTE, TDCC (EDIA), UCS, UNTDI/TRADACOMS, and VICS
- Multiple versions and releases of standards
- Electronic standards distribution to speed delivery of new standards
- Ability to migrate a map from one version/release of a standard to another or from one transaction to another
- Online creation and customization of standards
- Full standards compliance checking (user option)

Additional Features

- Designed for high throughput and performance
- Support for concurrent users and applications
- Support for shared Trading Partners profile (Minimal Trading Partners)
- Ability to process in multiple environments
- VSAM or DB2 implementation available
- MVS or CICS implementation
- Support for encryption and authentication
- Application program interface (API) to integrate directly with your application
- Utilizes the Security Access Facility (SAF) to establish system security down to the record level
- Graphical User Interface to simplify the management of profiles, standards, data formats, and maps
- Support for MQSeries Queues

Overview of DataInterchange

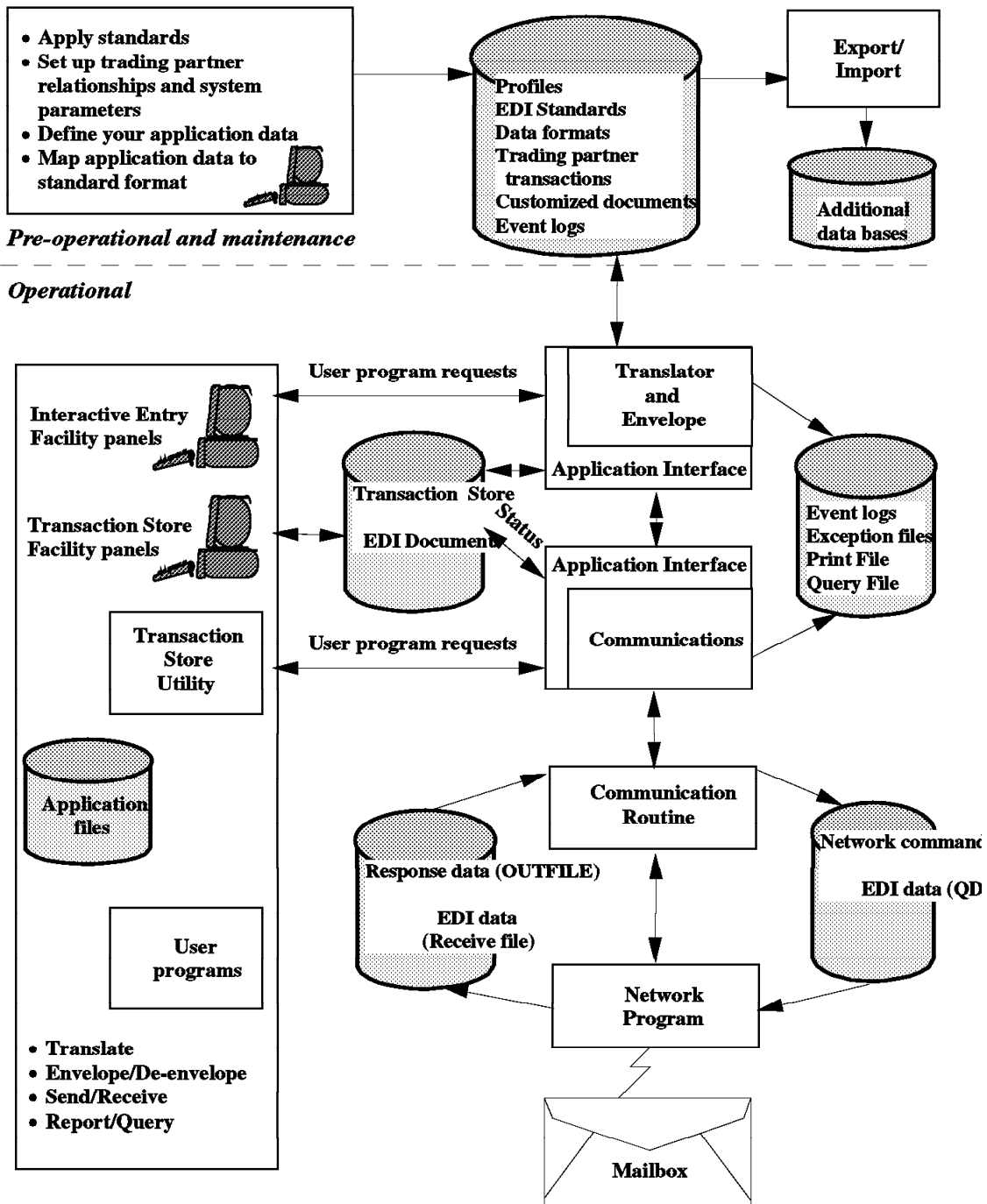


Figure 1-2. DataInterchange Services Overview

Before You Begin

DataInterchange must be installed in environments created for your use. See the *DataInterchange Installation Guide* and the *DataInterchange Programmer's Reference* for more information.

In addition, your system's security administrator must have granted you access to DataInterchange resources. If you are the security administrator, see Appendix A, "Security," for instructions.

Setup for Sending and Receiving Data

Note: If you are using the DataInterchange Client interface, see "Using DataInterchange with the DataInterchange Client Graphical User Interface" on page 1-13 and the *DataInterchange Client User's Guide* for additional information.

Before you can use DataInterchange to send or receive transactions, messages or files, certain information needs to be defined within DataInterchange. This information is used to describe how data is to be sent or received, how it is to be formatted in your application files and mapped to a standard, who is to receive the data or from whom it is to be received, and other pertinent information. The following outlines the necessary set-up steps:

1. Install and establish your environment

Install DataInterchange in MVS or CICS. See the *DataInterchange Installation Guide*.

2. Add the EDI and envelope standards

Select the EDI standards you wish to use and add them to your DataInterchange database. See Chapter 3, "Importing and Requesting EDI Standards Electronically" for additional information.

You may customize the EDI standards you have applied now or at a later time. Customizing can be helpful to alter a standard to suit a particular business need. See Chapter 6, "Customizing EDI and Envelope Standards."

3. Verify/create a network profile

Network profiles are used to define which networks you will use to communicate with your trading partners. The network profile includes information such as the network ID and the communications routine you will be using. DataInterchange contains profiles for the IBM Global Network and GEIS network. See "Setting Up the Network Profile" on page 5-4.

4. Define yourself to DataInterchange

Requester profiles are used to identify each group or individual in your organization who will be sending and/or receiving data using DataInterchange. Examples of the information in this profile are the requestor's network and network address. See "Setting Up the Requestor Profile" on page 5-14.

5. Define your trading partners

A trading partner profile can be created for each trading partner with whom you will do business.

Trading partners can share a profile using the Minimal Trading Partner feature. The profile contains information used to identify the trading partner as well as other trading partner specific data.

Examples of the information in this profile are the trading partner's network and network address. See "Setting Up the Trading Partner Profile" on page 5-23.

6. Customize the envelope profile

DataInterchange provides default envelope profiles to supply information for envelope header and trailer records. You need to customize the sender ID field of the envelope profile you will be using. If you use more than one sender ID, you need to create an additional envelope profile for each sender ID you use. See "Envelope Profiles" on page 6-30 for additional information.

7. Define your application data format to DataInterchange

Application data formats are used to define the format of your application data to DataInterchange. Several record formats are supported by DataInterchange. One application data format can be used to describe input application data that is sent to a trading partner and output application data that is received from a trading partner. See Chapter 7, “Defining and Working With Your Application Data,” for additional information.

8. Map your application data to a standard

The application data format you defined to DataInterchange in the previous step now needs to be mapped to a standard. This means associating the fields in your application data to the data elements in a standard. You will create a map for sending data and/or one for receiving data. See Chapter 9, “Mapping Your Application Data to an EDI Standard Transaction Set.”

As part of mapping, you may choose to specify translation and/or validation tables, as needed. Translation tables allow you to substitute one value for another when sending or receiving transactions. Validation tables verify that a piece of data is contained in a list of acceptable values. If not, a translation error will occur. See Chapter 8, “Translation and Validation Tables,” for additional information.

9. Identify which trading partners will use the maps

As part of mapping, you define which trading partners will use each map. Maps also define information used when sending or receiving a transaction, such as encryption keys, functional acknowledgment information, the type of envelope to use and the ID of the envelope profile. Maps may be used by multiple trading partners. See “Add Trading Partner Usage for Sending” on page 9-78 and “Add Trading Partner Usage for Receiving” on page 9-88.

Sending Your Data to a Trading Partner

The following outlines how to send your application data to a trading partner.

1. Translate application data to a standard
2. Envelope the translated data
3. Send the enveloped data

Translating Application Data to Standard

The first step in sending your data to a trading partner is to invoke DataInterchange to translate your data to a standard format. Your data can be placed in a file before invoking DataInterchange or it can be passed directly to DataInterchange.

As your application data is translated, it is placed in the Transaction Store and can be managed online via the *Transaction Store Facility*. For information on using this facility, see Chapter 10, “Managing Your EDI Data Using the Transaction Store Facility.”

Errors encountered during translation processing result in error messages being placed in the log and print files. In addition, the application data involved in the error will be placed in the exception file.

Collecting Application Data for Translation

Typically, one or more applications place application data into one or more application files over a duration of time. Periodically, a task is invoked to request the DataInterchange Utility to translate the data in one or more of these application files to a selected EDI standard. Multiple tasks may be invoked simultaneously or at different times to translate the application data.

Translating Application Data as it is Generated

An application may also place application data into a file and then invoke the DataInterchange Utility or the DataInterchange API directly to request the application data be translated to a selected standard. In this scenario, the application data can consist of one or more transactions. Multiple applications invoking DataInterchange may be running simultaneously.

See the *DataInterchange Programmer's Reference* for additional information on translating application data, the DataInterchange Utility, and the Application Programmers Interface.

Enveloping and Sending Transactions

When application data has been translated successfully and placed in the DataInterchange Transaction Store, it must be enveloped. Enveloping involves placing the EDI standard header and trailer around your transactions in preparation for sending. Your transactions are placed into an intermediate file as they are enveloped. When the transactions have been enveloped, they can be sent to your trading partner.

To send your enveloped data, DataInterchange invokes a communications routine that passes the enveloped data in the intermediate file to a network program it invokes. The network program is supplied by you or another provider, such as a value added network (VAN). The network program sends the enveloped data over the network you selected for communications to your trading partner.

The translation, envelope, and send functions may be requested individually or they may be combined as needed. You can:

- Translate
- Envelope
- Send
- Translate and Envelope
- Envelope and Send
- Translate, Envelope, and Send

The envelope and send functions can be invoked as a task separate from your application or they can be invoked by your application. See the *DataInterchange Programmer's Reference* for additional information on the DataInterchange Utility and the application program interface. In addition, the Envelope and the Envelope and Send functions may be performed via the Transaction Store Facility. See Chapter 10, "Managing Your EDI Data Using the Transaction Store Facility," for additional information.

Collecting translated application data in the transaction store and enveloping and sending it together in one envelope at a later time may help to reduce networking costs, instead of sending many envelopes to each trading partner.

Envelope Overrides

DataInterchange provides you with the capability to override particular values in the interchange envelope with data provided by your application. A default envelope will be selected if an override is not needed.

Envelope overrides may be provided in the following ways:

- One of the steps in the DataInterchange trading partner definition is to define a trading partner usage. The trading partner usage is used to select the correct mapping to be used to translate data. An envelope override may be specified on the trading partner usage.

For outbound processing, the envelope profile member specified may contain constant values which will override values generated during the envelope process. The profile member is used to generate the interchange header and trailer segments.

For inbound processing, the envelope profile member specified is used to generate the interchange header and trailer segments for a functional acknowledgment (if requested).

For more information, see “Specifying Trading Partners” on page 9-78.

- Application data format definition is one of several steps in the DataInterchange trading partner transaction mapping process. This definition specifies records and files in your application data file. DataInterchange currently provides two definitions for the application data (Raw data and Control and Data (C & D) record format). Envelope overrides may be specified on the C record. More information can be found in the *DataInterchange Programmer's Reference*.
- DataInterchange provides several methods to process your data through translation. The Application Programming Interface (API) is one method that may be used to couple your application program with the DataInterchange translation process. To use the API, your application must use the predefined control blocks to communicate information to the translation process. One of these control blocks is the translator control block (TRCB). The TRCB allows your application to specify envelope overrides used to generate the interchange headers and trailers. More information can be found in the *DataInterchange Programmer's Reference*.

Receiving Data From Your Trading Partner

Steps similar to sending data are involved in receiving data from a trading partner.

1. Receive the standard data
2. Deenvelope the standard data
3. Translate the standard data to an application format

Receiving and Deenveloping Transactions

Periodically, poll the network(s) you use to see if there are transactions waiting to be received. Receiving standard transactions involves invoking DataInterchange and requesting the *receive* function.

DataInterchange calls its communications routine which invokes the network program supplied by you or another provider. If transactions are received, they are placed into an intermediate file. You can specify all transactions be received and/or only transactions from a certain trading partner, depending on the network you use.

After transactions have been received, they can be deenveloped. Deenveloping removes the envelope headers and trailers from the standard transactions contained in the intermediate file and places the results in the Transaction Store. Functional acknowledgments are also generated if necessary. Once transactions are in the Transaction Store, they can be managed using the Transaction Store Facility or translated to application format and processed by your application. See Chapter 10, “Managing Your EDI Data Using the Transaction Store Facility.”

The receive and deenvelope functions can be invoked as a task separate from your application or they can be invoked by your application. See the *DataInterchange Programmer's Reference* for additional information on the DataInterchange Utility and the application program interface. In addition, the receive and deenvelope function can be performed using the Transaction Store Facility.

Translating Standard Data to Application Format

You can invoke DataInterchange to translate certain transactions contained in the Transaction Store to your application format. Transactions translated to your application format will be placed in an application data file, where they accumulate until your application processes them.

The translate function can be invoked as a task separate from your application or it can be invoked by your application. See the *DataInterchange Programmer's Reference* for additional information on the DataInterchange Utility and the application program interface. In addition, the translate function can be performed using the Transaction Store Facility.

The receive, deenvelope, and translation functions can be requested individually or they can be combined as needed. You can:

- Receive
- Deenvelope
- Translate
- Receive and Deenvelope
- Deenvelope and Translate
- Receive, Deenvelope, and Translate

Continuous Receive

With DataInterchange for CICS, you can invoke a continuous receive function. This function allows you to continually receive data from one or more trading partners as soon as the data is delivered to your mailbox. DataInterchange continuous receive functions will receive data as it enters your mailbox, deenvelope it, translate it to an application data format, and invoke an application response routine to process the data. Processing to be done in a continuous receive mode is specified in a continuous receive profile. See “Continuous Receive Facility” on page 5-43 and the *DataInterchange Programmer's Reference* for additional information.

Translating Application Data to Application Data

DataInterchange for MVS and DataInterchange for CICS provide the ability to translate data from an application data format to a standard format, and from an application format to a different application data format. When application data is translated from one application data format to another, the source file (containing the From application data) and the target file (containing the To application data) are required. You define the target file as a standard, using the cReate standard facility command, and then create a map to direct the movement of data from the source application format to the target application format. This map is created in the same way that a map that translates an application to a standard is created. For additional information on how to implement application data to application data translation, see “Translating Application Data to Application Data (Fixed-to-Fixed)” on page 7-23.

Functional Acknowledgments

For some transactions you send, you may want the receiver to reply, acknowledging the receipt, acceptance, or rejection of the transaction. Such a reply is called a functional acknowledgment. A functional acknowledgment indicates only that a specified set of envelopes was syntactically correct or incorrect. A functional acknowledgment does not indicate that the business agrees to the business terms contained in the envelopes. For more information, see “Requesting, Sending, and Receiving Functional Acknowledgments” on page 5-48.

Functional acknowledgments are automatically created, if required, during deenveloping. They are placed in the Transaction Store and optionally enveloped to be ready for sending. The DataInterchange envelope function can be used to envelope functional acknowledgments at a later time. Use the DataInterchange send function to transmit functional acknowledgments to the network and your trading partner. Your task or application can issue the send function if immediate turnaround of functional acknowledgments is desired.

Update Status

If the transactions you send are supposed to generate network acknowledgments, you will need to issue an update status request periodically. This can be done using the Transaction Store Facility, the DataInterchange Utility, the DataInterchange application program interface, or the Continuous Receive Facility. See Chapter 10, “Managing Your EDI Data Using the Transaction Store Facility” and the *DataInterchange Programmer’s Reference* for additional information on the DataInterchange Utility and the application program interface.

DataInterchange Reporting on Your EDI System

DataInterchange supplies several reporting and auditing tools to help you manage your EDI system. These tools include online inquiries and displays, formatted reports, data extracts, and exception reports.

Online Inquiries Using the Transaction Store Facility

The Transaction Store Facility allows you to obtain information and status about transactions and envelopes that were sent to or received from your trading partners.

Formatted Reports Using the DataInterchange Utility

You can use the DataInterchange Utility to specify search criteria and produce the following reports:

1. Activity Summary

Creates a summary of your inbound and outbound transaction activity. This summary includes the total number of outgoing and incoming transactions that meet your search criteria. On the outbound side, it categorizes the transactions into the number of translated transactions, enveloped transactions, and sent transactions. On the inbound side, it categorizes the transactions into the number of transactions translated and those not yet translated.

2. Acknowledgment Image

Returns functional acknowledgment images for transactions that meet your search criteria.

3. Transaction Details

Creates detailed information about individual transactions that meet your search criteria. This includes information about your trading partner, application data, acknowledgments, and networks that relate to the individual transactions selected.

4. Transaction Image

Returns transaction images for EDI documents that meet the search criteria.

5. Status Summary

Returns summary information about individual transactions that meet your search criteria. This information includes the transaction handle, trading partner nickname, data format ID, transaction

status, store status, date enveloped, interchange control number, network status, group control number, and functional acknowledgment status.

6. Status Summary2

Includes all the information from Status Summary plus the application control number and the internal trading partner ID value for each transaction.

7. Event Log

Returns all entries from the DataInterchange Event Log that meet your search criteria.

8. Query

Returns a list of transactions, identified by a date and time stamp called the transaction handle, that meet your search criteria.

For more information, see the *DataInterchange Programmer's Reference*.

Transaction and Envelope Data Extracts Using the DataInterchange Utility

Transaction and Envelope data extracts allow you to retrieve information from the DataInterchange Transaction Store. You can then format the information retrieved to meet your business needs. A Transaction data extract could be used to determine what documents have not been functionally acknowledged by your trading partners. An envelope data extract could report on which applications were generating transactions to reconcile system usage.

For more information, see the *DataInterchange Programmer's Reference*.

Management Reporting Data Extracts

Management Reporting data extracts include information about trading partner profiles and capabilities, data maps and their usage, network activity, and transaction statistics. Management Reporting data extracts can be used to answer questions such as how many purchase orders were sent to each of your trading partners last month, or which trading partners you are trading purchase orders with at the X12 V2R3 level. DataInterchange provides the following data extracts to gather this information:

- Trading Partner Profile Data Extract

This data extract provides complete identification information about your trading partners, such as company name address, contacts, telephone number, nickname, network name, interchange ID, account ID, and user ID. Additionally, the date of the last transmission of data is provided including the interchange, group, and transaction control numbers.

- Trading Partner Capability Data Extract

This data extract provides a subset of trading partner identification information and detailed information regarding the cumulative transactions exchanged with a trading partner, such as mapping direction, standards used, transaction ID, measurement date range, total number of transactions processed, and number of transactions that had errors.

- Network Activity Data Extract

This data extract provides a record of DataInterchange's network activity. You can use this information to verify the integrity of your EDI process, and to understand and reconcile network charges. Data provided includes the network ID, name and account number, user ID, direction, charge code, control date, and the total number of interchange envelopes and characters of data sent or received.

- Transaction Activity Data Extract

This data extract provides a record of daily transaction activity for individual trading partners. You can use this information to verify the integrity of your EDI process, and to understand and reconcile network charges. Trading partner data provided includes a subset of trading partner identification information and detailed information regarding the daily transactions exchanged with a trading partner. Transaction data provided includes mapping direction, standard used, transaction ID, map ID, application data format ID, measurement date, total number of transactions processed, and number of transactions that had errors. Outbound functional acknowledgments will not be reported through Transaction Activity Data Extract Reports.

All data extracts are formatted as sequential files that contain fixed length records. For more information about Management Reporting data extracts, see the *DataInterchange Programmer's Reference*.

Migrating Maps

DataInterchange allows you to automatically migrate your maps from one version/release of a standard to another. For example, you can migrate a map from EDIFACT 90.2 to EDIFACT 91.2. You can also use this to migrate a map from one transaction set to another. For example, you could migrate a map from an X12 850 to an X12 860.

For more information, see “Migrating a Transaction Mapping” on page 9-97 or the *DataInterchange Programmer's Reference*.

Export/Import

You can move your maps, profiles, application data formats and more from one DataInterchange system to another:

- From a test to a production system using export/import
- From a production system to a test system using export/import
- To exchange data with other DataInterchange users
- To move data between a DataInterchange client and a DataInterchange host
- To migrate to a higher release level

For more information, see Chapter 11, “Exporting and Importing Transactions,” or the *DataInterchange Programmer's Reference*.

Interactive Entry Facility (IEF)

You can create EDI documents and send them to your trading partners interactively using IEF. You can also send and receive free form messages and files interactively.

For more information, see Chapter 13, “Using the Interactive Entry Facility (IEF).”

Using DataInterchange with the DataInterchange Client Graphical User Interface

This section provides an overview of the DataInterchange Client interface and some considerations when using DataInterchange Client. For complete documentation on the DataInterchange Client interface, refer to the *DataInterchange Client User's Guide*.

DataInterchange Client is a Windows-based, graphical user interface that can be used to do many of the tasks that were done previously with the host interface. With DataInterchange Client, you can take advantage of the capabilities of Windows and graphical techniques, which make tasks much easier to perform.

DataInterchange Client replaces many, but not all, of the functions of the DI host interface. You can use DataInterchange Client to:

- Create and update your operational profiles.
- Create and update your trading partner profiles.
- Create and update envelope profiles.
- Create and update your trading partner contacts.
- Load and maintain standards.
- Define your application data formats.
- See a graphical side-by-side view of your data formats and standards as well as the mapped relationships between the two.
- Perform mapping of data formats to standards using drag and drop actions.
- Generate control strings (compile your maps).
- Preview or print predefined reports for all of your profiles, maps, data formats, and so on.
- Create, save, and run queries to display only the information important to you.
- Obtain status information from the Transaction Store (client/server mode only).
- Obtain the event log information.

In addition, this interface can be used to:

- Edit items from different databases.
- Edit items in other windows without exiting the one you are working on.
- Copy data with ease from one database to another.
- Change the standard or data format associated with a map at any time without performing a Migrate Mapping function.
- Repeat qualifications and mappings with ease.
- View uses by a trading partner.
- Document extensive comments associated with your profiles, maps, data formats, and so on.
- See the user ID, date, and time associated with the last update to an item.

- Customize the way data is displayed to you.
- Maintain a list with extensive information about all your contacts and associate a specific contact with multiple trading partner profiles.
- Create queries to allow you to find items with given attributes (for example, find Trading Partners in Florida).

ATTENTION:

DataInterchange Client does not perform EDI translation. It is a 32-bit graphical user interface to the DataInterchange for MVS and DataInterchange for CICS translator products.

Client/Server Mode versus Standalone Mode

DataInterchange Client can be used in either client/server mode or standalone mode. Client/server mode means there is a direct connection between the interface on the PC and the DB2 databases on the host. Updates made on the PC client are immediately reflected in the host databases as they are saved.

For client/server mode, you must have client/server middleware in place for a realtime connection between DataInterchange Client and the host DB2 databases. You can use the DataInterchange Client interface to create and update profiles and trading partner information. These changes are made directly to the host database. You can access the host standards, data formats, and maps from DataInterchange Client and convert them to the new DataInterchange Client format. Then, you can use DataInterchange Client to update the data formats, standards, and maps on the PC and compile (generate) a new control string to be placed in the host database.

In standalone mode, there is no direct connection between the DataInterchange Client graphical user interface and the host databases. In this mode, you first export the object from the DataInterchange host (that is, export a trading partner or a map). The export file is downloaded to the PC and imported into DataInterchange Client. After changes are made to the object you export from DataInterchange Client, upload to the host, and import back into DataInterchange host. If you are working with Setup profiles or Trading Partner profiles, the updated profile is what you export and upload to the host. If you are working with Standards, Data Formats, or Maps, then only the compiled control string is exported and uploaded to the host. Standalone mode allows you to utilize the advantages of the graphical user interface to create and maintain objects without the need for a client/server middleware software package.

Note: You should have controls to prevent users from updating the same objects from two different interfaces. Without these controls, a user could overwrite changes made by another user. The DataInterchange host was modified to provide control information to warn users when they try to update a map or control string that belongs to the DataInterchange Client. The following table identifies possible values for the new **Type** field that appears on Transaction Mappings panel (TP01).

Table 1-1 (Page 1 of 2). Values for Type Field

Value	Meaning	Description
blank		The map and control string were last updated on the host.
C	Converted	The host map was converted to the DataInterchange Client format and a new control string was generated on the DataInterchange Client. The original map still exists on the host; however, it is not the latest version. If you try to update the host map, you will receive a warning message.
I	Incomplete Conversion	The host map was converted to the DataInterchange Client format, but the control string was not regenerated. The translator still uses the original control string generated from the host map. If you try to update the host map, you will receive a warning message.

Table 1-1 (Page 2 of 2). Values for Type Field

Value	Meaning	Description
P	PC Map	The map only exists on the DataInterchange Client.

DataInterchange Client Databases

DataInterchange Client was designed to be Open Database Connectivity (ODBC) compliant, so that it can be used with a variety of PC databases as well as with the host DB2 databases. ODBC compliance also allows DataInterchange Client to interface with a variety of client/server middleware products.

Naming Convention Changes for DataInterchange Client

In the DataInterchange Client interface, some of the terminology has been changed to reflect generally accepted conventions as well as requests from customers. Please note the following changes:

Table 1-2. DataInterchange Terminology Changes

DI Host Terminology	DataInterchange Client Terminology
Requestor Profiles	Mailboxes
Security Profile	Network Security
Network Operation Profile	Network Commands
Application Definition Profile	Application Defaults
System Profile (Persistent Environment)	CICS Performance
Trading Partner Transactions (TPT)	Maps
Validation Tables	Code Lists
Translation Tables	Forward and Reverse Translation Tables
Application Data Formats (ADFs)	Data Formats
Multi-Occurrence Structure Mapping	Path Qualified Mapping
Generates a Control String from a Map	Compiles a Map into a Control String
Standard Envelope Data	Envelope Defaults

Unavailable Functions in the DataInterchange Client Interface

The following functions are unavailable in the DataInterchange Client interface and must still be accessed through the DataInterchange host interface.

- Interactive Entry Facility (IEF)
- Maintenance of language profiles
- DataInterchange Utility
- Translation and communication options of the Transaction Store

Moving to DataInterchange Client

DataInterchange Client is designed so you can move from the host to the client at your own pace. You may choose to move all your objects at one time, or move them one at a time. This section provides an overview of the steps you take to begin updating DataInterchange objects using the DataInterchange Client interface. Some objects can be accessed and updated immediately, such as setup profiles and trading partner profiles. Others must first be converted from the host format to the new DataInterchange Client format, such as standards, data formats, and maps. For details on the conversion process, refer to the *DataInterchange Client User's Guide*.

Note: Objects that you wish to move to DataInterchange Client must originate from the DataInterchange Version 2 Release 1 or higher host product. You cannot move objects from earlier releases directly to DataInterchange Client. You must first import them to a DataInterchange Version 2 Release 1 or higher host. This is normally done during the installation of the DataInterchange host product.

Moving Setup Profiles and Trading Partner Profiles

Setup profiles and trading partner profiles can be accessed from DataInterchange Client without first converting the profiles. Setup profiles include mailboxes, activity logs, network profiles, and so on.

Client/Server mode

If you have client/server access to the compatible DataInterchange host databases, you can create new profiles or update existing ones using the DataInterchange Client windows. Changes you make using the DataInterchange Client are immediately reflected in the host databases when you save your changes.

Standalone mode

To update an existing profile, follow these steps:

1. Export the profile from the DataInterchange Version 2 Release 1 or higher host.
2. Download the profile to the PC where DataInterchange Client is running.
3. Import the profile into DataInterchange Client.
4. Update the profile using the DataInterchange Client windows.
5. Export the profile from DataInterchange Client.
6. Upload the file to the host.
7. Import the profile into DataInterchange Version 3 Release 1.

Note: Another alternative is to export all profiles at one time and maintain them using DataInterchange Client. When you make a change to a profile, export the new profile to the host.

Moving Standards, Data Formats, and Maps

Client/Server mode

Standards, data formats, and maps must be converted to a new format before they can be used by the DataInterchange Client interface. When they are converted, they should be maintained by the DataInterchange Client interface. They should not be moved back to the host without careful consideration.

To move a standard, data format, or map to DataInterchange Client in client/server mode, follow these steps:

1. In DataInterchange Client, use the Conversion Browser to view lists of the standards, data formats, or maps on the host.

2. Highlight the item you want to convert.
3. Click the Convert option from the Actions file menu.

As a result, the item will be accessible from the DataInterchange Client.

Standalone mode

To update an existing standard, data format, or map, follow these steps:

- | 1. Export the item from DataInterchange Version 2 Release 1 or higher host.
2. Download the item to the PC where DataInterchange Client is running.
3. Import the item into DataInterchange Client.
- | 4. Use the Open Browser option, then the Conversion option, to view lists of the standards, data formats, or maps in the import file.
5. Highlight the item you want to convert.
6. Click the Convert option from the Actions file menu.
7. Update the item using the DataInterchange Client windows.
8. Compile the map (generate the control string).
9. Export the compiled map from DataInterchange Client.
10. Upload the compiled map to the host.
- | 11. Import into DataInterchange Version 3 Release 1.

Chapter 2. Getting Started with DataInterchange

Accessing DataInterchange for MVS	2-1
Logon Options for DataInterchange for MVS	2-1
Logging On to DataInterchange for MVS	2-3
Accessing DataInterchange for CICS	2-3
Logon Options for DataInterchange for CICS	2-3
Logging On to DataInterchange for CICS	2-3
Selecting from the Menu	2-5
Using DataInterchange Panels	2-6
Panel Areas	2-6
Hierarchy of User Actions	2-8
Entering Commands at the Command Prompt	2-8
Entering Data	2-9
Initial Cursor Position	2-9
Requesting Help (F1)	2-9
Scrolling and Using the List Action	2-10
Printing Reports and Lists	2-11
Exit (F3) and Cancel (F12)	2-11
Prompt (F4)	2-11
Extended Prompt for Mapping and Application Data Formats	2-14

Chapter 2. Getting Started with DataInterchange

This chapter describes how to sign on to DataInterchange for MVS and DataInterchange for CICS, and describes the DataInterchange panels.

Before you can sign on to DataInterchange, a security administrator must give you access to DataInterchange resources. If you are the security administrator, see Appendix A, "Security," for instructions.

Accessing DataInterchange for MVS

Logon Options for DataInterchange for MVS

The logon command for DataInterchange for MVS is EDI unless the person who installed DataInterchange for you chose another name for this command. You can enter just the command and accept the default options in the EDI CLIST, or you can specify up to six logon options to override the default options. The logon options for DataInterchange for MVS are:

Option	Description
System ID	Identifies the copy of DataInterchange you are accessing. If more than one copy of DataInterchange is installed on your system, for example, a test version and a production version, the System ID option indicates which copy you want to access. Specify the option as shown: SYSID(system-name) where: <i>system-name</i> is the resource name the security administrator assigned to the copy you are accessing. The default is DIENU.
Print	Directs printed output for the session, except for print requests you issue from the Transaction Store Facility. Your print options are: PRINT(PRINTER) Prints the information immediately after you log off. PRINT(HOLDQ) Sends the information to the spool hold queue. You use a system facility to print the information after you log off. This option is the default supplied in the EDI CLIST. PRINT(FILE) Uses a physical file named <i>userid.EDI.PRTFILE</i> for output. If this file does not exist, the logon command creates it for you. You use a system facility to print the information after you log off.
Report file	Directs reports and printouts that you request while using the Transaction Store Facility. Your report file options are: RPTFILE(PRINTER) Prints the information immediately after you log off.

- RPTFILE(HOLDQ)** Sends the information to the spool hold queue. You use a system facility to print the information after you log off. This option is the default supplied in the EDI CLIST.
- RPTFILE(FILE)** Uses a physical file named *userid.EDI.RPTFILE* for output. If this file does not exist, the logon command creates it for you. You use a system facility to print the information after you log off.

Language profile Indicates which language variables to use for the session. The Language profile option sets the language variables for the session. For example, one of the language variables is the format for entering and displaying dates. Specify the option as shown:

LANGPROF(member-name)

where:

member-name
is the name of the language profile member that contains the variables you want to use. The default is ENU (US English).

For more information about the language profile, see “Language Profile (LANGPROF)” on page 4-3.

Language ID The Language ID option determines which language version is used for the session. Version 1 Release 4 and higher releases are only available in US English. The language ID is ENU. Enter this option as shown:

LANGID(ENU)

Export/import file Indicates which export/import files to use for the session.

The export/import file options are:

- EIFILE(Y)** Uses a physical file named *userid.EDI{EIFILE* for output (export) or input (import). If this file does not exist, the logon command creates it for you. This is the default.
- EIFILE(N)** Allows you to specify separate files for different types of export or import data. The format of a file name is:

userid.EDI.Elxxx

where:

xxx

is one of the following:

File	Description
STD	Standards
ADF	Application data formats
TPT	Trading partner transactions
CST	Control strings
DDF	Business document definitions
MAP	Business document layouts
PRF	Profiles
TBL	Tables

If any of these files do not exist, the logon command creates them for you. Using separate files can speed processing.

If you want to use all of the default options in the EDI CLIST, enter **EDI** to logon to DataInterchange.

If you want to use a test system and send printed output to the system printer when you end the session, you can log on to DataInterchange using the following:

EDI SYSID(TEST) PRINT(PRINTER)

Default values in the EDI CLIST are used for the options you did not specify with the logon command.

You can also set the logon options by changing the defaults in the EDI CLIST, but this changes the defaults for all users.

Logging On to DataInterchange for MVS

To log on to DataInterchange for MVS, type the logon command **EDI**, followed by any options you want to use for the session, and press Enter. The DataInterchange Administrator's Menu (MP01) is displayed.

```
MP01                      DataInterchange MVS Version 3.01 Main Menu

                          5655-B29 (c) Copyright IBM Corp. 1989, 1998
                          All Rights Reserved.  Licensed materials - Property of IBM

Type the number of your choice and press Enter, or press the Exit key to
exit.

Choice ==>  —  1. Interactive Entry Facility
                2. Profiles
                3. Event Logging
                4. Trading Partner Transactions (Mapping)
                5. EDI Standards
                6. Application Data Formats
                7. Translation and Validation Tables
                8. Envelope Standards
                9. Transaction Store Facility
               10. Export
               11. Import
               12. Utility

Command ==>
Enter Tso F1=Help F3=Exit F9=Retrieve F12=Cancel
F13=Keys help
```

Accessing DataInterchange for CICS

Logon Options for DataInterchange for CICS

The CICS transaction to access the DataInterchange for CICS panels is EDIA. Release 4 and higher releases are available only in US English. The language ID is ENU. You can use just the command to access DataInterchange for CICS.

Logging On to DataInterchange for CICS

To log on to DataInterchange for CICS, type the transaction name **EDIA** and the language ID, if necessary, and press Enter. The DataInterchange Administrator's Menu (MP01) is displayed.

```
MP01                      DataInterchange for CICS Version 3.01 Main Menu

                          5655-B30 (c) Copyright IBM Corp. 1989, 1998
                          All Rights Reserved.  Licensed materials - Property of IBM

Type the number of your choice and press Enter, or press the Exit key to
exit.

Choice ==>  _  1. Interactive Entry Facility
                2. Profiles
                3. Event Logging
                4. Trading Partner Transactions (Mapping)
                5. EDI Standards
                6. Application Data Formats
                7. Translation and Validation Tables
                8. Envelope Standards
                9. Transaction Store Facility
               10. Export
               11. Import

Command ==>
Enter  F1=Help  F3=Exit  F9=Retrieve  F12=Cancel
F13=Keys help
```

Note: All panels in DataInterchange for MVS have TSO in the function key area. In DataInterchange for CICS, this option is not supported or displayed.

Selecting from the Menu

When you log on to DataInterchange, the first menu displayed is the DataInterchange Administrator's Menu (MP01) Main Menu.

```
MP01                      DataInterchange for MVS Version 3.01 Main Menu

                          5655-B29 (c) Copyright IBM Corp. 1989, 1998
                          All Rights Reserved.  Licensed materials - Property of IBM

Type the number of your choice and press Enter, or press the Exit key to
exit.

Choice ===>  _  1. Interactive Entry Facility
                2. Profiles
                3. Event Logging
                4. Trading Partner Transactions (Mapping)
                5. EDI Standards
                6. Application Data Formats
                7. Translation and Validation Tables
                8. Envelope Standards
                9. Transaction Store Facility
               10. Export
               11. Import
               12. Utility

Command ===>
Enter Tso F1=Help F3=Exit F9=Retrieve F12=Cancel
F13=Keys help
```

Some of the choices shown on this panel may not appear on your menu. Only the functions that you were given access to by the security administrator are listed. For example, your menu may display only the following options:

```
MP01                      DataInterchange for MVS Version 3.01 Main Menu

                          5655-B29 (c) Copyright IBM Corp. 1989, 1998
                          All Rights Reserved.  Licensed materials - Property of or Licensed to IBM

Type the number of your choice and press Enter, or press the Exit key to
exit.

Choice ===>  _  1. Interactive Entry Facility
                2. Profiles
                4. Trading Partner Transactions (Mapping)
                6. Application Data Formats

               10. Export
               11. Import
               12. Utility

Command ===>
Enter Tso F1=Help F3=Exit F9=Retrieve F12=Cancel
F13=Keys help
```

To see a description of the choices, press F1 (Help). To return to the menu, press F3 (Exit). To select an option, type the number of that option in the **Choice** field, and press Enter.

Using DataInterchange Panels

Panel Areas

When you select an option from the Administrator's Menu (MP01), the applicable panel is displayed. For example, the Profile Definitions panel (PM01) is displayed when you select **Profiles**.

Log work with Members Print View A			
PM01		Profile Definitions	1 to 17 of 17
Action	Profile ID	Profile Description	Log?
—	ACTLOGS	Activity log	N
—	ADAMCTL	User program information	N
—	APPDEFS	Application Definition profile	N
—	CONTRECV	Continuous Receive Profile	N
—	E	EDIFACT standard envelope data	N
—	I	ICS standard envelope data	N
—	LANGPROF	Language profile	N
—	MQSERIES	MQSeries Queue Profile	N
—	NETOP	Network operation profile	N
—	NETPROF	Network profile	N
—	REQPROF	Requestor profile	N
—	SECUPROF	Security profile	N
—	SYSPROF	System profile	N
—	T	UN/TDI standard envelope data	N
—	TPPROF	Trading partner profile	N
—	U	UCS standard envelope data	N
—	X	X12 standard envelope data	N

C
D

Command ==> E
 Enter F1=Help F3=Exit F7=Bkwd F9=Retrieve
 F12=Cancel F13=Keys help

The panel ID (PM01) is shown near the upper-left corner, and the panel title is centered on the same line.

Most DataInterchange panels include five areas:

- A** Action bar The top line of the panel lists the commands or actions that you can select on the panel you are viewing. The commands or actions available on the Profile Definitions panel (PM01) are:

Command/Action	Mnemonic
Log	L
work with Members	M
Print	P
View	V

The command/action mnemonic is the capitalized letter in the command/action name. This is usually the first letter, with exceptions, such as the action work with Members, whose mnemonic is M. On some terminals, the mnemonic is underlined as well as capitalized. You can enter the mnemonic in uppercase or lowercase.

You choose an action by entering its mnemonic in the action column adjacent to the item with which you want to work. On some panels, Action is abbreviated to A as the column heading.

For example, if you enter a V, the mnemonic for View, in the first entry field (ACTLOGS) and press Enter, the profile definition for the Activity Log is displayed.

F3 (Exit) or F12 (Cancel) returns you to the Profile Definitions panel (PM01).

If there are several items you want to work with, you can type an action for each item before pressing Enter. Each action you select is processed. If an error occurs before all the actions are processed, an error message is displayed, and the remaining actions are not processed until you correct the error and press Enter for the remaining actions, which are still displayed. Actions that were processed do not remain displayed on the panel.

- B

Panel body

The middle of the screen is the working area, which contains entry fields, lists of selectable items, menu choices, and scrollable text.

C

Message area

This is the line directly above the command prompt. Status messages and error messages are displayed in this area. Serious error messages and messages that do not fit on one line are displayed in a pop-up window.

D

Command prompt

The command prompt (Command ==>) provides an alternative method of entering your request. You can enter the same commands at the command prompt that you enter using the action column or the function keys. For example, while in profile, you can enter the following command at the command prompt to view the profile definition for the Activity log.

V ACTLOGS

The profile definition for the Activity log is displayed.

- E

Function keys

This is the area below the command prompt. The active function keys for the panel are listed in this area. Panels that appear as windows in full-screen panels do not have a separate function key area and action bar. The function keys and actions that appear on the background panel apply to the current window. The definitions of the function keys remain the same throughout DataInterchange. You can use F13 (Keys help) to display a list of all function key definitions, similar to the following:

Key	Command	Description
Enter	TSO	Accepts input data
F1-Help		Enter TSO command
F3-Exit		Shows online help
F4-Prompt		Ends the current task
F7-Backward		Displays additional information
F8-Forward		Shows the previous page
F9-Retrieve		Shows the next page
F12-Cancel		Recalls previous command
F13-Keys help		Goes back one panel
F19-Previous		Shows help for keys
F20-Next		Shows the previous occurrence
		Shows the next occurrence

On the actual list, the active keys are highlighted.

Notes:

1. The TSO function allows you to enter a TSO command, program, or a CLIST name from the command prompt. Type **TSO** followed by the TSO function you want to invoke, and press Enter. When you exit TSO, the DataInterchange panel is redisplayed.
2. The TSO function is not supported or displayed in DataInterchange for CICS.

Hierarchy of User Actions

Priorities exist between the actions taken through the use of function keys, entries on the command line, and entries in the action column. DataInterchange performs the actions in the following order:

1. Function keys
2. Command line
3. Action column

For example, you can:

1. Enter an action in the action column
2. Enter a command at the command prompt
3. Press the Help key

Because the function keys have first priority, the help panel is displayed. The action and command are not processed. When you exit from the Help function, the entries are still displayed. When you press the Enter key, processing action begins. The command entry is processed first, according to priority, followed by the action entry processing. If there are multiple actions entered, the actions are processed sequentially, with the exception of the List and Line actions, which are always processed last.

Entering Commands at the Command Prompt

When you enter text at a command prompt in the form of a command or a command abbreviation, the system processes the text to determine which command is associated with the text entered. The following steps process the command:

1. If the text is a single character, and an action bar is included on the panel, all mnemonics for the current panel are searched for a match.
2. All of the command literals defined for the product are searched for a match. The match can be either exact or partial.
 - If an exact match for the command literal is found and the command is available on the current panel, the command is executed.
 - If an exact match is found but the command is not available on the current panel, a message is displayed indicating that the command is not available.
 - If a partial match is found for a single command literal, the command is either executed or the COMMAND NOT AVAILABLE message is displayed.
 - If more than one partial match is found for two command literals entered at the command prompt, and both of the matches are listed as available, a message indicating that the text matches more than one command is displayed.
 - If more than one partial match is found, and only one of the partial matches is for the command available on the panel, that command is executed.

Entering Data

Underscoring on the panels indicates where to type entries and how long the entries can be. For example, on the Update Profile Member panel (PM10), the fields are followed by data and underscores. The underscore indicates the field is an entry field, and length indicates how many characters you can enter.

To change existing values in an entry field, type your entry over the existing data and erase any remaining characters.

The value in any field whose name ends with a colon (:) cannot be changed. This type of field usually appears near the top of the panel; for example, see the **Profile ID** and **Network ID** fields on the following Update Profile Member panel (PM10).

PM10	Update Profile Member	1 to 15 of 15
Profile ID . . . :	NETPROF	
Network ID . . . :	IINB41	
Network name . . .	IBM Global Network	_____
Communication rtn	VANIINB1	
Network program . .	IEBASE	__
Network parameters	_____	
Network input file	INMSG	__
Input rec length	80	__
Trans data queue	QDATA	__
Trans rec length	80	__
Time zone	GMT	__
System type	_____	
System level . . .	_____	
Msg text header . .	T	
Net output file . .	OUTMSG	__
Message handler . .	INB1MSG	__
Network sequence	00000	

Some panels display information in columns. Some columns are for information only and cannot be changed. Underscores indicate where you can enter data.

Use the Tab key to move from field to field. To submit your data to the system and go to the next panel, press Enter. To discard your entries, press F3 (Exit) or F12 (Cancel) before you press Enter.

Initial Cursor Position

An information-only panel is a panel that does not have an entry field in the panel body. The initial cursor position for all information-only panels is the upper-left corner. For any panel that is not an information-only panel, the cursor is positioned at the first entry field in the panel body.

If you access and then exit from a Help panel, the cursor is returned to its previous location. However, if the cursor was anywhere in the command prompt area, it is returned to the first position in the command prompt area.

Requesting Help (F1)

You can request help for the following items:

Item	Action
Entry fields	To display an explanation of a field, place the cursor on the field and press Help.
Action bar choices	To obtain help for action bar choices (commands), place the cursor on a command in the action bar and press Help.
Function keys	<p>There are two methods to obtain help for function keys:</p> <ul style="list-style-type: none"> • Place the cursor on a function key in the function key area and press Help. • Press F13 (Keys help). Place the cursor on an active (highlighted) function key in the list and press Help. Additional information is displayed about the function key.
Displayed messages	To obtain more information about a displayed message, place the cursor on the message and press Help. The help information displayed explains the message or how to correct the error. A message number is included in the Help title. You can use the message number to look up the message in <i>DataInterchange Messages and Codes</i> , or to report a problem to support personnel. Message help cannot be requested for any message that is displayed for a help panel.
Panels	<p>General help for a panel is displayed if the cursor is not positioned:</p> <ul style="list-style-type: none"> • In an entry field • On an action bar choice • On a function key • On a displayed message <p>To access general help for the panel, move the cursor to an undefined area in the panel body and press F1 (Help).</p>

When the available help information does not fit on one panel, a scroll indicator appears in the upper-right corner. The scroll indicators are a plus (+) and a minus (-) symbol. The plus symbol indicates you can scroll forward to see more information. The minus symbol indicates you can scroll backward.

To exit from the Help panel and return to the previous panel, use the F12 (Cancel) function key.

Scrolling and Using the List Action

Scrolling allows you to view additional lines of information that did not fit on a panel. When additional information is available, the scroll function keys, Backward (F7) and Forward (F8), are displayed in the function key area. You can scroll forward and backward using these function keys.

When scrolling through a long list, you may want to use the List action to view the additional data. List makes the item of your choice the first item in the scrollable list. You cannot scroll backward when you use the List action; you must use List again to display preceding items.

There are two methods for using List:

- Tab to the item you want placed at the top of the list, type **L**, and press Enter.
- At the command prompt, type **L** and the ID of the item you want at the top of the list. The ID does not have to be exact; the program finds the nearest match.

Note: When you enter the List action in combination with other actions, by default the List action is processed last.

Printing Reports and Lists

The Print and Report actions appear in the action bar if a list or report is available from the panel you are using. For example, if you are using the Profile Definition panel, you can use Print to get a list of the members in a profile. The help for the Print action explains what information is printed.

DI for MVS writes the output of these actions to the data definition names (ddname) PRTFILE and RPTFILE. DI for CICS writes the output to the temporary storage (TS) queue EDIP`tttt`, where `tttt` is the CICS terminal ID.

Exit (F3) and Cancel (F12)

The F3 (Exit) and F12 (Cancel) function keys allow you to halt a task or exit a panel:

- F3 (Exit) ends the current task.
- F12 (Cancel) returns you to the previous panel.

If you are only one panel deep into the task, either function key ends the current task. If you have typed data, but have not pressed Enter, the data is lost. Repeated use of the F3 (Exit) or F12 (Cancel) function keys returns you to the menu.

Use either the F3 (Exit) or F12 (Cancel) function key to return to a previous panel from a Help panel.

The F3 (Exit) function key is also used to stop multiple actions. For example, if you have typed multiple View actions and decide you do not want to see all of the items, you can use the F3 (Exit) function key to discard the remaining View actions.

Prompt (F4)

The F4 (Prompt) function key allows you to fill in entry fields by selecting from a list of values that are valid for that particular field. Prompt gives you the opportunity of recognizing and selecting the choice you want, rather than having to remember all the choices and typing the desired one. For example, you may want to use Prompt to select a Trading Partner from a list of all Trading Partners.

If Prompt is available for an entry field, a plus (+) symbol will appear after the field. For example, on the Trading Partner and Network Criteria panel (TF04), F4 (Prompt) can be used on the following fields:

- Trading partner nickname
- Network ID
- Network status
- Network ack pending (Y/N)

TF02Criteria Selections

The list of envelopes or transactions you will work with depends on the selection criteria you enter. To see the list based on default criteria for the task you chose, press Enter. To limit the list with specific criteria

TF04Trading Partner and Network Criteria

TTrading partner nickname
7TInternal trading partner ID
E
-ANetwork ID
-SDate sent
Time sent
Network status
Network ack date
Network ack time
Network ack pending (Y/N)

+
+
to
to
+
to
to
+

Command ==>

Enter TS0 F1=Help F3=Exit F4=Prompt F9=Retrieve
F12=Cancel F13=Keys help

To use F4 (Prompt), you would place the cursor on an entry field that is followed by a plus (+) symbol and press F4. A prompt list panel will appear containing a single choice list. For example, if you place the cursor on the Network ID field and press F4, the following panel is an example of the results.

TF02Criteria Se

The list of envelopes or transactions selection criteria you enter. To see for the task you chose, press Enter. criteria

TF04Trading Par

TTrading partner nickname
7TInternal trading partner ID
E
-ANetwork ID
-SDate sent
Time sent
Network status
Network ack date
Network ack time
Network ack pending (Y/N)

+
+
to
to
+
to
to
+

Network IDMore: +

- GEIS
- IIN
- IINB1
IINB41
- IINCICS
- IINR3

Command ==>

Enter TS0 F1=Help F3=Exit F4=Prompt F8=Fwd F9=Retrieve
F12=Cancel F13=Keys help

To select a Network ID that you want, type a slash (/) in the action column next to the appropriate Network ID, and press Enter. The Network ID will then be placed in the entry field and the Prompt window will be closed, as shown in the following panel.

TF02
Criteria Selections

The list of envelopes or transactions you will work with depends on the selection criteria you enter. To see the list based on default criteria for the task you chose, press Enter. To limit the list with specific criter

TF04
Trading Partner and Network Criteria

T
Trading partner nickname
+

7
T
Internal trading partner ID

E
Network ID
IINB41
+

-
A
Date sent
to

-
S
Time sent
to

Network status
+

Network ack date
to

Network ack time
to

Network ack pending (Y/N)
+

You can qualify the prompt list by entering a mask in the entry field before pressing F4 (Prompt) or after the prompt list is displayed by changing the mask and pressing the Enter key. A mask is any combination of characters or wild card symbols (ie, * or ?). The asterisk (*) matches any string of characters. The question mark (?) matches one single character. For example, if you want to see all of the Network IDs that started with IIN, you would enter IIN* and press F4 (Prompt). When using this example, the following screen would be the result:

TF02
Criteria Selections

The list of envelopes or transactions you will work with depends on the selection criteria you enter. To see the list based on default criteria for the task you chose, press Enter. To limit the list with specific criter

TF04
Trading Partner and Network Criteria

T
Trading partner nickname
+

7
T
Internal trading partner ID

E
Network ID
IIN*
+

-
A
Date sent
to

-
S
Time sent
to

Network status
+

Network ack date
to

Network ack time
to

Network ack pending (Y/N)
+

Network ID
- IIN
- IINB1
- IINB41
- IINCICS
- IINR3

If you want to see all of the Network IDs that ended in CICS, change IIN* to *CICS and press Enter. When using this example, the following screen would be the result:

TF02
Criteria Selections

The list of envelopes or transactions you will work with depends on the selection criteria you enter. To see the list based on default criteria for the task you chose, press Enter. To limit the list with specific criter

TF04
Trading Partner and Network Criteria

T
Trading partner nickname
+

7
T
Internal trading partner ID
+

-
E
Network ID
+

-
A
Date sent

-
S
Time sent

Network status

Network ack date

Network ack time

Network ack pending (Y/N)
+

Network ID

IINCICS

Extended Prompt for Mapping and Application Data Formats

There is an extended prompt feature for entry fields for which you would enter an application field name or application structure name. This feature will allow you to display the Application Data Format complete with structures, fields, and levels of indentation. This will be very useful during the mapping process to find the correct fields for mapping elements and structures for qualifying loops. You will be able to switch between an alphabetized list of either structures or field names and the data format prompt display.

TP10
Map Data Element - BEG01

Transaction ID : TESTS
Loop ID . . . :
Segment ID . . : BEG

Standard definition of data element
Element ID . . : 353 Elem qualifier :
Sequence num . : 01 Description . : Transaction Set Purpose
Type : ID Req des . . . : M
Minimum length : 002 Maximum length : 002

Definition of data element for this mapping transaction
Application field name . . +
Literal (enter on next line)

Accumulator/action . . . / _ / _ / _ / _ / _
Special handling (Y/N) . . N +

If you are on the Map Data Element panel (TP10) and you place the cursor on the Application field name field and press F4 (Prompt), the Application field name panel is displayed with a list of application field names defined in the Application Data Format definition associated with this Trading Partner Transaction (Mapping). The field names are listed in alphabetical order.

Data format

TP10

Transaction ID : TESTS
Loop ID . . . :
Segment ID . . : BEG

Standard definition of data
Element ID . . : 353
Sequence num . : 01
Type : ID
Minimum length : 002

Application field name

More: +

D

COMMENT

- DESCRIPTION

- PARTNUMBER

- PODATE

- PONUMBER

- QUANTITY

- TOTALITEMS

- UNITOFMEASURE

Definition of data element

Application field name . .

Literal (enter on next line)

+

Accumulator/action

Special handling (Y/N) . .

N +

/

-

/

-

/

-

/

-

The Application field name panel allows you to display the Application Data Format definition in a variety of ways. The Data format command that is listed in the Application Action Bar can be used to display the list of structures defined in the Application Data Format definition. The structure level is displayed with an indentation of one space for each level. To display the Application field name panel with a list of structures, enter a **D** in the action column. An example of the results is shown on the next panel.

With the Application Data Format structures listed, you can use the Expand command to expand a structure and display all the fields and structures with that structure. The Expand command only expands the outer level structure. It does not expand the structures defined within the outer level structure, unless it is entered on the command line. For example, if you enter the Expand command beside TUTORBASE, it does not expand the LINEITEMS structure. If the Expand command is entered on the command line, then all of the structures in the list are expanded.

The first time the Data format prompt panel is displayed for a particular Application Data format, only the list of structures and their level of indentation is shown. You can use the Expand command to expand a structure and display all of the fields and structures within that structure. The Expand command only expands the outer level structure. It does not expand structures defined within the outer level structure unless it is entered on the command line. For example, if you enter the Expand command beside TUTORBASE, it does not expand the LINEITEMS structure. If the Expand command is entered on the command line, then all of the structures in the list are expanded.

You can use the Fields command to switch back to the alphabetized list of field names. If you were requesting Prompt for an entry field that required a structure name, then the command to switch back to the list of structures would be Structures.

ExpandFields

TP10Map Data Element - BEG01

Transaction ID : TESTS
Loop ID . . . :
Segment ID . . : BEG

Standard definition of data element
Element ID . . : 353Elem qualifier :
Sequence num . : 01Description . : Transaction Set Purpose
Type : IDReq des . . . : M
Minimum length : 002Maximum length : 002

Definition of data element for this mapping transaction
Application field name . . +
Literal (enter on next line

Application field name

A Type Name
E Str TUTORBASE
- Str LINEITEMS

Accumulator/action
Special handling (Y/N) . .

Command ==>

An E is entered beside TUTORBASE to expand it, and the following screen would be an example of the result:

CollapseExpandFields

TP10Map Data Element - BEG01

Transaction ID : TESTS
Loop ID . . . :
Segment ID . . : BEG

Standard definition of data element
Element ID . . : 353Elem qualifier :
Sequence num . : 01Description . : Transaction Set Purpose
Type : IDReq des . . . : M
Minimum length : 002Maximum length : 002

Definition of data element for this mapping transaction
Application field name . . +
Literal (enter on next line

Application field name

A Type Name
- Str TUTORBASE
- Fld PODATE
- Fld PONUMBER
- Str LINEITEMS
- Fld TOTALITEMS
- Fld COMMENT

Accumulator/action
Special handling (Y/N) . .

Command ==>

If at least one structure in the list has been expanded, a new command, Collapse, is now on the Application Action bar. This command will allow you to collapse a previously expanded structure. The Collapse command will collapse all levels of a structure. This will remove the display of all field names that belong to primary and underlying structures. For example, if the TUTORBASE and the LINEITEMS structures had been expanded, and you entered a C in the action column beside TUTORBASE, then both

TUTORBASE and LINEITEMS would be collapsed. If this command is entered on the command line, then all the structures in the list are collapsed.

Collapse
Expand
Fields

TP10
Map Data Element - BEG01

Transaction ID : TESTS
Loop ID . . . :
Segment ID . . : BEG

Standard definition of data element
Element ID . . : 353
Elem qualifier :
Sequence num . : 01
Description . : Transaction Set Purpose
Type : ID
Req des . . . : M
Minimum length : 002
Maximum length : 002

Definition of data element for this mapping transaction
Application field name . . +
Literal (enter on next line

Application field name

/

Accumulator/action
Special handling (Y/N) . .

A
Type
Name

C
Str
TUTORBASE

-
Fld
PODATE

-
Fld
PONUMBER

-
Str
LINEITEMS

-
Fld
TOTALITEMS

-
Fld
COMMENT

Command ==>

To select an Application field name, enter a slash (/) beside the desired field name. If you enter a mask while on the Data Format prompt panel, it will be ignored. It is ignored because this is not an alphabetized list of one type of data; it is a list of fields and structures that are shown in the layout of an Application Data Format with relationships and indentation levels. The next time you request F4 (Prompt) for this Application Data Format, the same panel you were last on will be shown unless you enter a mask. If you enter a mask, then you will get the alphabetized list of fields. Position within the Data Format panel is remembered between invocations of F4 (Prompt) so that you will not have to expand the structures again.

Chapter 3. Importing and Requesting EDI Standards Electronically

Importing EDI and Envelope Standards	3-1
Importing New Standards	3-1
Obtaining Standards through Electronic Distribution	3-2
Program Prerequisites	3-2
Requesting New Standards	3-3
Allocating Your Data Sets	3-6
DataInterchange for MVS	3-6
DataInterchange for CICS	3-6
Receiving New Standards	3-7
DataInterchange for MVS	3-7
Considerations	3-7
DataInterchange for CICS	3-7

Chapter 3. Importing and Requesting EDI Standards Electronically

This chapter describes how to add electronic data interchange (EDI) standards to your DataInterchange database using the DataInterchange Standards Import facility. It also describes how to obtain approved standards electronically.

EDI standards define the segments and data elements that make up a transaction set such as a purchase order or an invoice. An EDI standard contains one definition for each unique segment and data element. These segments and data elements are then used in as many transaction sets as necessary.

Envelope standards define the segments and data elements that make up the headers and trailers enclosing transaction sets, functional groups, and interchanges. EDIFACT refers to envelope standards as service segments, and X12 refers to them as envelope interchange control segments. Envelope standards also specify the default delimiters used in standard data, such as the data element separator, component data element separator, and segment terminator.

Each EDI standard specifies an envelope standard. Although it is recommended that you use the specified envelope standard, you can use a different envelope standard or customize the specified envelope standard to meet your needs.

Importing EDI and Envelope Standards

When you install DataInterchange, you receive a copy of the standards currently approved by the primary standards organizations. The standards are shipped in the DataInterchange Export/Import tagged format, which must be imported to either the VSAM or DB2 repository, depending on your environment. Each standard is separately loaded during installation into EDI.V3R1M0.EISTDS. This file serves as your primary standards source and should not be modified.

To add a standard to your database, use the DataInterchange Import function. By importing only the standards you require, you improve performance and reduce storage requirements and maintenance.

Importing New Standards

You can import the standards interactively or in the background. If you use interactive import, start DataInterchange using the following parameter after the CLIST name: **STDADM(y)**. This indicates to DataInterchange Import function that EDI.V3R1M0.EISTDS is the import file to be used. If no standards are found in EDI.V3R1M0, or you encounter problems with the file, refer to the *DataInterchange Installation Guide* to verify that you completed loading the standards correctly.

For more information about the import facility, see Chapter 11, "Exporting and Importing Transactions."

DataInterchange provides a utility for importing files in batch mode. For DataInterchange for MVS, the command syntax is:

PERFORM IMPORT WHERE CTLFILE(ddname)

where:

ddname

is a batch control file that describes the contents of the data being imported.

Importing/Requesting EDI Standards

| For DataInterchange for CICS, the command syntax is:

PERFORM IMPORT WHERE CTLFILE(*name*) CTLTYPE (*type*)

where:

name

is the name of a temporary storage queue (TS), transient data queue (TD), or VSAM entry sequenced data set (VS); *type* indicates the file type of *name*: TS, TD, or VS.

For more information about using the DataInterchange Utility to import standards or for sample JCL, see the sections “Using the DataInterchange Utility” and “Using Sample JCL” in the *DataInterchange Programmer’s Reference*.

Obtaining Standards through Electronic Distribution

Updates to standards are provided and maintained as part of DataInterchange. Electronic distribution of standards expedites the receipt of new standards. Once a standard is approved and available on magnetic media from the International Organization for Standardization (EDIFACT) or the American National Standards Institute (X12), it will be available through electronic distribution in approximately 60 days.

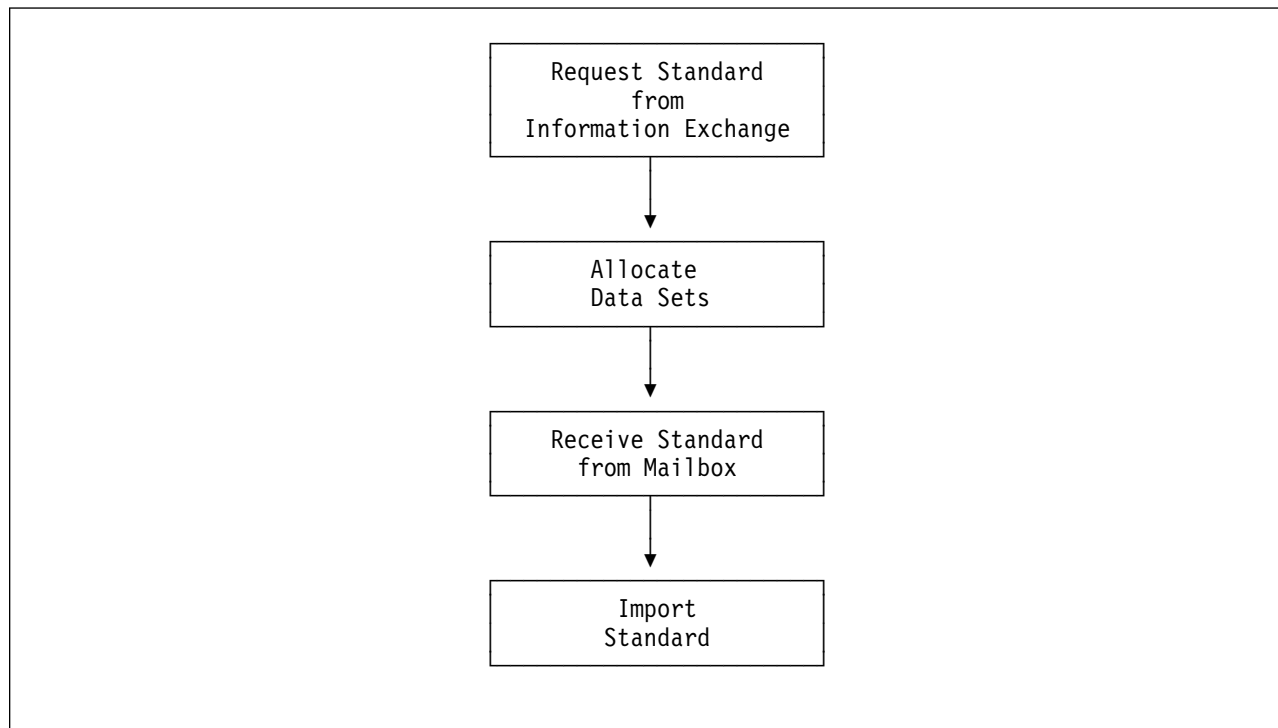


Figure 3-1. Obtaining Standards Electronically Flowchart

Program Prerequisites

| To receive standards using electronic distribution, you can use Information Exchange. The IBM Global
| Network user ID you will be using to request standards must have access to the Information Exchange
| Administrative Services product. You can also download standards from the DataInterchange product web
| site. See Appendix C, “The IBM DataInterchange Web Site,” for more information.

Requesting New Standards

To request new standards, follow these steps:

1. Log on to Information Exchange Administration Services. The Information Exchange Administration Services Main Menu (M0000US) is displayed.
2. Type **7** on the command line, and press Enter.

```

M0000US          Information Exchange Administration Services
                  Main Menu
                  System ID : USF

Action  _  Work with one of the following:
           1. Profiles
           2. Alias tables
           3. Distribution lists
           4. Messages
           5. Audit trails and session traces
           6. Trading partners
           7. Libraries
           8. Events
           9. X.400 services
          10. Carbon copy relationships
          11. System programmer services
           X. Exit from Information Exchange Administration Services

(C) IBM 1983, 1992; Advantis 1993, 1997; IBM 1999

Command ==> 7

```

The Work with Libraries panel (M7000US) is displayed.

3. Type **C** in the **List Format** field for a list of all libraries. Tab the cursor to the command line, type **1**, and press Enter.

```

M7000US          Work with Libraries

Account ID . . . . . ACCT
User ID . . . . . USERID
Owning account ID . . . . .
Library . . . . . (actions 2, 3, 4, 5, 6, 7, or 8)
Member . . . . . (actions 6, 7, or 8)
List format (C/A) . . c (action 1)

Action . . . . . - 1. List libraries
                   2. Add a library
                   3. Delete a library
                   4. Change a library
                   5. List library members
                   6. Add or replace a library member
                   7. Delete a library member
                   8. Retrieve a library member
                   9. Search library members

Command ==> 1
Enter F1=Help F3=Exit F4=Main Menu

```

The List Libraries panel (M7100US) is displayed.

Importing/Requesting EDI Standards

4. To start the list at a specific library, type **ATLD** if you are in the United States or Canada; **USATLD** if you are in Europe; **LTLD** if you are in Latin America, Mexico, New Zealand, and Australia. Press Enter. (This service may not be available on the Information Exchange System in Japan.)

M7100US

List Libraries

Account ID ACCT

User ID USERID

Locate account/library.. **atld** (press Enter)

Codes (multiple selection):

C = Change library information

D = Delete library

K = Search library members

M = List library members

S = See additional information

Code	Account	Library	Description (truncated to a length of 44)	Read	Write
-	AAAA	BBBBB	Ccc	Y	Y
-	XXXX	YYYY	Zzz	N	N

The specified library is now at the top of the list.

5. Type **M** in the code column on the line containing PRODUCT SUPPORTED EDI STANDARDS, and press Enter.

M7100US

List Libraries

Account ID ACCT

User ID USERID

Locate account/library.. (press Enter)

Codes (multiple selection):

C = Change library information

D = Delete library

K = Search library members

M = List library members

S = See additional information

Code	Account	Library	Description (truncated to a length of 44)	Read	Write
M	ATLD	DISTAND	DI/MVS CICS: PRODUCT SUPPORTED EDI STANDARDS	Y	
-	ATLD	DISUPPT	DI/MVS CICS: GENERAL SUPPORT INFORMATION	Y	N
-	ATLD	DIKITS	DI/MVS CICS: INDUSTRY SOLUTION KITS	Y	N
-	ATLD	DISTANDT	DI/MVS CICS: PRODUCT SUPPORTED EDI STANDARDS	Y	N

The List of Library Members panel (M7500US) is displayed.

There are two libraries that contain EDI standards stored in DI import file formats. DISTAND library members contain standards that are stored in NATIVE format. DISTANDT library members contain standards that are stored in TAGGED format. New standards will only be added to DISTANDT library members.

The library member README is updated regularly to provide you with information regarding the status of new or unreleased standards. You can view the README library member by typing **V** in the code column next to the member, and pressing Enter. If the README library member is not available in your country’s Information Exchange Administration Services facility, contact your country representative and request this service.

6. Type **R** in the code column next to each standard you want to retrieve, and press Enter.

```

M7500US                               List Library Members

Account ID . . . . . ACCT
User ID . . . . . USERID
Owning account ID . . . . . ATLD
Library . . . . . DISTAND

Start at member . . . . . (press Enter)
Codes (multiple selection):
  D = Delete member                R = Retrieve member
  S = See additional information    V = View member text (BILLABLE)

Code  Member  Description (truncated to a length of 60)
  R  EDI911    UN/EDIFACT STATUS 1 - 1991 91.1
     EDI912    UN/EDIFACT STATUS 2 - 1991 91.2
     ODETTEV3   ODETTE - Version 3 WITH 1991/1992 CORRECTIONS
     -  README  Status of Unreleased Standard
     -  TDCCV2R9 TDCC Version 2 Release 9
     -  UCSV3R2  UCS Version 3 Release 2
     -  X12V3R2  ANSI X.12 Version 3 Release 2
     -  X12V3R22 X12 VERSION 3, RELEASE 2, SUBRELEASE 2
  
```

The Retrieve a Library Member panel (M7800US) is displayed.

7. In the Receiver's information area, type the account ID and user ID you want the system to send the standard to, and although the panel says press Enter to exit the panel, press F3 (Exit) to process your request and exit the panel.

```

M7800US                               Retrieve a Library Member

Account ID / User ID . . . . . ACCT / USERID
Owning account ID / Library . . . . . ATLD / DISTAND
Member . . . . . EDI912

Receiver's information (Fill in only one line)
  Sys ID - Account ID - User ID . . . . . - acct - userid
  List name . . . . .
  Alias type - table - name . . . . . - -
  Message charge (REQUIRED) (1, 3, 5, or 6) . 6
  Message name. . . . .
  Message sequence . . . . .
  User message class . . . . .
  Acknowledgments
    (A/B/C/D/E/F/R/blank) . . . . . -
  Delivery class (I/P/blank). . . . .
  Retention period (Max 180). . . . . 030
  
```

Importing/Requesting EDI Standards

When the standard is placed in your mailbox, a message is displayed at the bottom of the List of Library Members panel (M7500US).

```
M7500US                               List Library Members

Account ID . . . . . ACCT
User ID . . . . . USERID
Owning account ID . . . . . ATLD
Library . . . . . DISTAND

Start at member . . . . . _____ (press Enter)
Codes (multiple selection):
  D = Delete member                      R = Retrieve member
  S = See additional information          V = View member text (BILLABLE)

Code  Member  Description (truncated to a length of 60)
-     EDI911  UN/EDIFACT STATUS 1 - 1991 91.1
-     EDI912  UN/EDIFACT STATUS 2 - 1991 91.2
-     ODETTEV3 ODETTE - Version 3 WITH 1991/1992 CORRECTIONS
-     TDCCV2R9 TDCC Version 2 Release 9
-     UCSV3R2  UCS - Version 3 Release 2
-     X12V3R2  ANSI X.12 Version 3 Release 2
-     X12V3R22 X12 VERSION 3, RELEASE 2, SUBRELEASE 2

0319 Member EDI912 from library DISTAND has been retrieved into the request
mailbox(es).
```

You can now return to the Administration Services Main Menu (M0000US).

Allocating Your Data Sets

DataInterchange for MVS

Allocate a new data set with the following attributes:

- Record Format: VB (Variable Block)
- Record Length: 4089
- Block Size: 4093
- Size: 2 megabytes

This data set will be used to store the new standard. If you are receiving one standard at a time, 2 megabytes is adequate; otherwise, allocate 2 megabytes for each standard you want to receive.

When starting DataInterchange, use the following parameter after the CLIST name: EDIEIFIL(*data set name*). This allocates the file as your Import or Export file.

DataInterchange for CICS

Verify that transient data queue **EDIS** is defined to the CICS region as intra-partition. If it is not, contact your CICS Systems Programmer to have this done.

Receiving New Standards

DataInterchange for MVS

There are two ways you can receive a standard from your mailbox:

1. Use Expedite communications programs.

Use Expedite Base/MVS communications program IEBASE to receive the requested standards from your IE mailbox. You can use sample JCL EDISTAND in library EDI.V3R1M0.SEDISAM1. See the appropriate Expedite product manual for additional information.

- ## 2. Use the Interactive Entry Facility.

For more information, see Chapter 13, “Using the Interactive Entry Facility (IEF).”

Each new standard file can take 30 minutes or longer to receive, depending on the size of the file and transmission speed.

Considerations

When you request multiple standards, separate files are sent to your IE mailbox. Receive the first standard and import it to DataInterchange before receiving and importing the second. Continue this sequence until all the requested standards have been received and imported. Although each file is sent with a different name, you must receive and import each file individually, because the data in the EIFILE data set is replaced by new data with each receive operation. If you do not receive and import standards individually, the data is lost. Alternatively, change the disposition of the allocation of the EIFILE file from OLD to MOD in the JCL, and ensure the file you allocated previously is large enough to hold all the standards you have requested. These changes allow you to receive all the standards at once.

DataInterchange for CICS

If you have DataInterchange installed in a CICS environment, follow these steps:

1. Sign on to Expedite/CICS using the LGO1 transaction identifier or its equivalent.
2. Type your account and user ID on the Expedite/CICS main menu (EXPMLG1), and press Enter.

```

EXPMLG1

0000
00
00 00 00 0000 0000 00000 00 000000 0000 * / 000 00 000 000
00 00 00 00 0 00 00 0 00 00 00 // 00 00 00 00
00 00 0000 0000 00 00 00 00 00 // 00 00 00 0000
00 00 00 00 00 00 0 00 00 00 // 00 00 00 00
0000 00 00 00 0000 00000 00 00 0000/ 0000 00 0000 0 00
0000 00 00 00 0000 00000 00 00 0000/ 0000 00 0000 000

Version 4.4.0
(C) Copyright IBM Corp., 1998. All rights reserved.
Licensed Material Property of or Licensed to IBM.

* Expedite is a trademark licensed to IBM.

Account < account > Userid < userid >

HI202 TYPE YOUR INFORMATION EXCHANGE ACCOUNT AND USERID, THEN PRESS ENTER.
Command ==>>
PF1=Help PF3=End

```

Importing/Requesting EDI Standards

The Expedite/CICS Information Session Start panel (EXPMMN1) is displayed.

If DataInterchange has already established a session with Expedite/CICS, the Expedite/CICS Main Selection Menu panel (EXPMMMS), shown in Step 4, is displayed instead.

3. Type your password, and press Enter.

```
EXPMMN1          Expedite/CICS    Information Exchange Session Start

Account          ACCT1
Userid           USER01

Current Information Exchange password:

Password          < password >

New password (optional):

New password      <           >

Verify new password <           >

HI209 TYPE YOUR INFORMATION EXCHANGE PASSWORD, THEN PRESS ENTER.

Command ==>
PF1=Help  PF3=End  PF12=CANCEL
```

The Expedite/CICS Main Selection Menu (EXPMMMS) is displayed.

4. Type **1** on the command line to work with receive data, and press Enter.

```
EXPMMMS          Expedite/CICS Main Selection Menu

Account          ACCT
Userid           USER01

Select the desired function:

1 REceive        Work with receive data
2 SEND           Work with send data
3 LIBraries      Work with libraries and members
4 LISt           Work with distribution lists
5 USErs          Work with user administration
6 SYStem         Work with system administration
7 PASS-through   Pass-through command(s) to Information Exchange
8 ALIas          Work with alias tables
E ESCape         Leave session ACTIVE and return to CICS
S STArt         Restart session with Information Exchange
X LOGoff         End session with Information Exchange

HI325 ENTER FUNCTION NUMBER, FIRST 3 LETTERS, OR =#. # ON THE COMMAND LINE.

Command ==> 1
PF1=Help  PF3=End
```

The Expedite/CICS Receive Selection Menu (EXPMRSM) is displayed.

5. Type **1** on the command line to issue a single receive, and press Enter.

```

EXPMSRM                      Expedite/CICS Receive Selection Menu

Account      ACCT
Userid       USER01

Select the desired function:

    1  SINGLe      Issue single receive
    2  STArT      Start continuous receive
    3  STOp       Stop continuous receive
    4  PRoCess    View a list of receives in process
    5  VIEw      View list of completed receives
    6  QUERy     Query the Information Exchange mailbox

    B  BATCh      Issue receive to process batch data
    S  SERvice    Issue receive for Electronic Customer Service

HI325 ENTER FUNCTION NUMBER, FIRST 3 LETTERS, OR =#. # ON THE COMMAND LINE.

Command ==> 1
PF1=Help  PF3=End
  
```

The Issue Single Receive panel (EXPSMI1) is displayed.

6. Complete the fields as shown on the panel below. However, replace ACCT with ATLD and USERID with ATLDSA with the account ID and user ID you used to request the standards from Information Exchange. Then press PF5 to receive the file.

```

EXPSMI1                      Issue Single Receive

Account      ACCT      From: System ID..... <      >
Userid       USERID   Account / Userid < acct > < user >
                                -or- List name..... <      >
                                -or- Alias name..... <      >

                                User class..... <      > (OPTIONAL)

To:
File name..... < edis > (File or Receive program name)
File management..... < td > (TS, TD, VS=VSAM, or PG=Program)
File type..... < b > (A=CRLF, B=LL, E=EDI, O=Other)
File disposition..... < N > (H=Hold, N=Normal process)
Retain header data.... < n > (Y - Keep headers / N - Discard)
                                (E - Use CDH and discard IE headers)
                                (F - Use CDH and retain IE headers)
Receive EDI..... <      > (Y=EDI only, N=NON-EDI, Blank=Both)
Handle records as..... < v > (S-Split / W-Wrap / V-Variable data)
Max record length..... < 00000 > (0000 - 28000 Characters)

HI308 ENTER FIELD INFORMATION AND PRESS PF5 TO RECEIVE MESSAGE(S).

Command ==>
PF1=HELP  PF5=Receive  PF12=Cancel
  
```

Importing/Requesting EDI Standards

7. Type **=E** on the command line, and press Enter, to exit Expedite/CICS while the receive is processed in the background. Avoid typing **=X** or you will log off Expedite/CICS and terminate the background receive processing. Because the new standard can take 30 minutes or longer to receive, periodically re-invoke Expedite/CICS to determine if the receive is complete. Because you did not log off, the Issue Single Receive panel (EXPMSI1) is redisplayed. The standard has been received when the following system message is displayed on this panel:

S0000002 SINGLE RECEIVE COMPLETED AT: *hh:mm:ss mm/dd/yy*

EXPMSI1		Issue Single Receive	
Account	ACCT	From: System ID..... <	>
Userid	USERID	Account / Userid < ACCT	> < USERID >
		-or- List name..... <	>
		-or- Alias name..... <	>
		User class..... <	> (OPTIONAL)
To:			
File name.....	< EDIS	>	(File or Receive program name)
File management.....	< TD	>	(TS, TD, VS=VSAM, or PG=Program)
File type.....	< B	>	(A=CRLF, B=LL, E=EDI, O=Other)
File disposition.....	< N	>	(H=Hold, N=Normal process)
Retain header data....	< N	>	(Y - Keep headers / N - Discard)
			(E - Use CDH and discard IE headers)
			(F - Use CDH and retain IE headers)
Receive EDI.....	< Y	>	(Y=EDI only, N=NON-EDI, Blank=Both)
Handle records as.....	< V	>	(S-Split / W-Wrap / V-Variable data)
Max record length	< 00000	>	(0000 - 28000 Characters)
HI308 ENTER FIELD INFORMATION AND PRESS PF5 TO RECEIVE MESSAGE(S).			
Command ==> =E			
PF1=Help PF5=Receive PF12=Cancel			

Chapter 4. Defining DataInterchange Operational Profiles

What is an Operational Profile?	4-1
Customizing National Language Support	4-2
Validation and Translation Tables	4-3
Single-Byte and Double-Byte Characters	4-3
Language Profile (LANGPROF)	4-3
User Program Information Profile (ADAMCTL)	4-5
MQSeries Queue Profile (MQSERIES)	4-7
Security Profile (SECUPROF)	4-9
Application Definition Profile (APPDEFS)	4-11
System Profile (SYSPROF)	4-14

Chapter 4. Defining DataInterchange Operational Profiles

This chapter provides information about defining operational profiles, such as language, user program information, and trading partner authorizations.

What is an Operational Profile?

A profile is a collection of descriptive information. It consists of a profile definition and one or more profile members. The profile definition provides the names of the data fields and their characteristics. The members are data records. For example, the trading partner profile contains one member or data record for each of your trading partners. Each member contains the same data fields. The first field of each member is the key or name of the member. For example, in the trading partner profile, the key is the trading partner nickname.

DataInterchange provides the profile definitions. It also provides members (data) for some profiles that you can use without modification. For other profiles that require modification, you must either add profile members or update those that are supplied.

Profile tasks include:

- Adding a profile member
- Copying and modifying a profile member
- Deleting a profile member
- Updating a profile member
- Printing a profile member or a list of members
- Viewing a profile definition or a profile member
- Turning logging on and off for a profile

When you select **Profiles** from the Administrator's Menu (MP01), the Profile Definitions panel (PM01) is displayed, listing the profile IDs in alphabetical order.

Log work with Members Print View			
PM01		Profile Definitions	1 to 17 of 17
Action	Profile ID	Profile Description	Log?
—	ACTLOGS	Activity log	N
—	ADAMCTL	User program information	N
—	APPDEFS	Application Definition Profile	N
—	CONTRECV	Continuous Receive Profile	N
—	E	EDIFACT standard envelope data	N
—	I	ICS standard envelope data	N
—	LANGPROF	Language profile	N
—	MQSERIES	MQSeries Queue Profile	N
—	NETOP	Network operation profile	N
—	NETPROF	Network profile	N
—	REQPROF	Requestor profile	N
—	SECUPROF	Security profile	N
—	SYSPROF	System profile	N
—	T	UN/TDI standard envelope data	N
—	TPPROF	Trading partner profile	N
—	U	UCS standard envelope data	N
—	X	X12 standard envelope data	N

Defining Operational Profiles

The profiles can be categorized into the following groups:

- Envelope header and trailer information

ID	Description
E	EDIFACT standard envelope data
I	ICS standard envelope data
T	UNTDI standard envelope data
U	UCS standard envelope data
X	X12 standard envelope data

For more information, see “Customizing Envelope Standards” on page 6-26.

- Network and mailbox information

ID	Description
NETPROF	Network profile. See “Setting Up the Network Profile” on page 5-4.
NETOP	Network operation profile. See the <i>DataInterchange Programmer's Reference</i> .
REQPROF	Requestor profile. See “Setting Up the Requestor Profile” on page 5-14.

- CICS requirements

ID	Description
CONTRECV	Continuous receive profile. This profile is available only with DataInterchange for CICS. See “Continuous Receive Facility” on page 5-43.

- Trading partner information

ID	Description
TPPROF	Trading partner profile. See “Setting Up the Trading Partner Profile” on page 5-23.

- Event logging and archiving

ID	Description
ACTLOGS	Activity log

For more information, see Chapter 12, “Event Logging.”

- User programs and options

ID	Description
ADAMCTL	User program information. See “User Program Information Profile (ADAMCTL)” on page 4-5.
APPDEFS	Application definition profile. See “Application Definition Profile (APPDEFS)” on page 4-11.
MQSERIES	MQSeries Queue profile. See “MQSeries Queue Profile (MQSERIES)” on page 4-7.
LANGPROF	Language profile. See “Language Profile (LANGPROF)” on page 4-3.
SECUPROF	Security profile. See “Security Profile (SECUPROF)” on page 4-9.
SYSPROF	System profile. See “System Profile (SYSPROF)” on page 4-14.

Customizing National Language Support

The logon options used to sign on to DataInterchange provide two parameters for controlling the use of language-specific resources and processing. The LANGID and LANGPROF parameters let you control which data sets and which language profile member are used for the session. For more information about defining the language parameters when you sign on to DataInterchange, see “Logon Options for DataInterchange for MVS” on page 2-1 or “Logon Options for DataInterchange for CICS” on page 2-3.

The language profile (LANGPROF) contains information that is considered unique for each language. You select the profile member to be used when you sign on. The code page ID field in the language profile determines the default code page used by DataInterchange. When the readable information (such as panels, help text, and messages) was created for your language, the information was created using the standard code page for that language. The code page ID field has been set to a specific standard code page.

Note: Version 1 Release 4 and higher releases are only available in US English.

DataInterchange gives this information to Graphical Display Data Manager (GDDM), which performs character conversion if the terminal being used is not the same code page. If this causes a problem, or if for some reason you do not want this to occur, you can prevent it by doing the following:

1. Copy the language profile member, creating a new member called NONE.
2. While copying the member, blank out the **code page ID** field. This turns off code page conversion.
3. Sign on to DataInterchange using **Langprof(None)** as a parameter.

You can also update this profile to change the displayed date format, time format, decimal notation, negative sign, and fold character. (Fold characters are described on page 4-5.) The language profile can contain as many members as necessary to satisfy your preferences.

Note: Version 1 Release 4 and later releases are available only in US English.

Validation and Translation Tables

DataInterchange provides three validation and translation tables that are directly related to customizing national language support. The character set table (CHARSET), alphanumeric table (ALPHANUM), and monospace table (MONOCASE) provide national language support as described in the following sections.

Note: Tables are described in Chapter 8, "Translation and Validation Tables."

Single-Byte and Double-Byte Characters

DataInterchange supports only single-byte characters for input data. This means that even if your language or terminal supports double-byte characters, you cannot type them in any data entry fields. (Double-byte characters are used in many Asian languages, such as Japanese.)

Note: If your application creates data that contains double-byte information, you must remove it before you attempt to process it with DataInterchange. If any transactions you attempt to send or receive contain double-byte information, severe errors can occur during the translation.

Language Profile (LANGPROF)

The language profile contains language variables, such as the formats you use for the date and time. Initially, it contains a member for the language version you install. ENU, for example, is the member for US English. You can add additional members or tailor some parts of the member that DataInterchange supplies.

When you sign on to DataInterchange, use the language profile option to indicate which member you want to use during the session.

Defining Operational Profiles

The following panel shows the definition of the language profile (LANGPROF).

PM05	View Profile Definition			1 to 7 of 7
Profile ID: LANGPROF		Profile description: Language profile		
	Field Label	Length	Type	Description
Key==>	Lang profile ID	006	CH	Language Profile ID (key)
	Code page ID	005	CH	Code Page ID
	Date mask	010	CH	Date edit/display mask
	Time mask	008	CH	Time edit/display mask
	Decimal notation	001	CH	Edit/display decimal notation
	Negative sign	002	CH	Preferred negative sign (display)
	Fold character	001	CH	Substitute for non-display char

Field

Lang profile ID

Description

Indicates the name of the member. For members supplied by DataInterchange, the name matches the 3-character language ID used for naming data files for different language versions. For example, ENU represents US English.

Code page ID

Indicates the language feature of DataInterchange that your company uses. For example, 00037 represents English.

Attention: Do not change this value. If you add members, they must contain either the same value as the member that was supplied or contain a blank code page ID to turn off code page conversion.

Date mask

Defines the format for entering, displaying, and printing dates. The mask can include the following keywords:

Mask Description

&C	Century and year (for example, 1998)
&Y	Year (for example, 98)
&M	Month
&D	Day

Any other character in the mask is a literal to be inserted in the same position it holds in the mask. Using November 26, 1998 as the date, two examples follow:

Mask	Date Format
&M/&D/&Y	11/26/98
&C &M &D	1998 11 26

Time mask

Defines the format for entering, displaying, and printing the time. The mask can include these keywords:

Mask Description

&H	Hour
&M	Minutes
&S	Seconds

Any other character in the mask is a literal to be inserted in the same position it holds in the mask. Using 59 minutes and 30 seconds after 11 pm, two examples follow:

Mask	Time Format
&H:&M:&S	23:59:30
&H.&M.&S	23.59.30

Field	Description								
Decimal notation	Indicates the character used in the decimal position for entering, displaying, and printing numbers. The translator recognizes this character as the decimal position in data you send and returns it in data you receive.								
Negative sign	Indicates the character or characters used for entering, displaying, and printing negative values. Use the first position for a leading sign or the second position for a trailing sign. Use an asterisk as a place holder in the unused position. For example: <table> <tr> <th>Mask</th><th>Description</th></tr> <tr> <td>-*</td><td>Leading minus sign, as in -123</td></tr> <tr> <td>*-</td><td>Trailing minus sign, as in 123-</td></tr> <tr> <td>()</td><td>Parenthetical notation, as in (123)</td></tr> </table> <p>The translator recognizes this negative sign in data you send and returns it in data you receive.</p>	Mask	Description	-*	Leading minus sign, as in -123	*-	Trailing minus sign, as in 123-	()	Parenthetical notation, as in (123)
Mask	Description								
-*	Leading minus sign, as in -123								
*-	Trailing minus sign, as in 123-								
()	Parenthetical notation, as in (123)								
Fold character	Indicates the character used to replace characters that cannot be displayed on the screen, such as end-of-line and carriage return.								

User Program Information Profile (ADAMCTL)

Members of this profile describe programs and exit routines you supply. Add a member to this profile for each of the following that you use:

Exit Routine or Program	Description
Field exit routine	Provide the name of this routine when mapping the data element.
Post-translate exit routine	Provide the name of this routine when defining the trading partner usage for a transaction you send.
Pre-translate exit routine	Provide the name of this routine when defining the trading partner usage for a transaction you receive.
Encryption exit routine	Provide the name of this routine in the security profile.
Authentication exit routine	Provide the name of this routine in the security profile.
Compression exit routine	Provide the name of this routine in the security profile.
Filtering exit routine	Provide the name of this routine in the security profile.
Monitor exit routine	Provide the name of this routine in the security profile.
Communication routine	Provide the name of this routine to handle communication between DataInterchange and a value-added network. The communication routines supplied by DataInterchange do not require a member in this profile.
Message processing program	Provide the name of this routine to handle responses from the network and update status information. The message handler programs supplied by DataInterchange do not require a member in this profile.

Defining Operational Profiles

Exit Routine or Program

Send and receive program for point-to-point connections

Description

Provide the name of this routine to issue network commands and process network responses for a point-to-point connection. This program is called by PTOPT, the communications routine for point-to-point connections.

Envelope exit routine

Provide the name of this routine to be called by DataInterchange to replace the standard read and write envelope processes.

You must add members for the programs you supply before referring to them in DataInterchange. For more information, see the *DataInterchange Programmer's Reference*.

The following panel shows the definition of the user program information profile (ADAMCTL).

PM05	View Profile Definition			1 to 15 of 15
Profile ID: ADAMCTL Profile description: User program information				
	Field label	Length	Type	Description
Key==>	Program name	008	CH	User program logical name
	Load module name	008	CH	User program physical name
	Program language	001	CH	Language used in user program
	Field Exit UE	001	CH	Userexit type is Field Exit (y n)
	Post-Translate UE	001	CH	Userexit type is Post-Translate
	Pre-Translate UE	001	CH	Userexit type is Pre-Translate
	Encryption UE	001	CH	Userexit type is Encryption
	Authentication UE	001	CH	Userexit type is Authentication
	Compression UE	001	CH	Userexit type is Compression
	Filtering UE	001	CH	Userexit type is Filtering
	Monitor UE	001	CH	Userexit type is Monitor
	Communication UE	001	CH	Userexit type is Communication
	Message Process UE	001	CH	Userexit type is Message Process
	Point-to-point UE	001	CH	Userexit type is Point-to-point
	Envelope UE	001	CH	Userexit type is Envelope

Field

Program name

Description

Indicates the logical name of your program.

- For a field exit routine or a pre-translate or post-translate exit routine, use the same name you use in the trading partner transaction.
- For an encryption, authentication, compression, or filtering exit routine, use the same name that is used in the security profile.
- For a communication routine, use the same name that is used in the network profile.
- For a send and receive program, use the same name you use for the network program in the network profile.
- For a message processing program, use the same name you use for the message handler in the network profile.
- For an envelope program, use the same name as specified in the IUSEREXIT field of the TRCB.

This program name is placed in the ZSNBNAME field of the SNB control block when the program is invoked.

Load module name Indicates the physical load module name that corresponds to the program name.

Field	Description										
Program language	Indicates the language in which the program is written:										
	<table> <tr> <th>Code</th><th>Program Language</th></tr> <tr> <td>A</td><td>Assembler</td></tr> <tr> <td>C</td><td>C language (IBM C/370 Compiler only)</td></tr> <tr> <td>K</td><td>COBOL programs compiled using the IBM COBOL II compiler</td></tr> <tr> <td>J</td><td>COBOL programs compiled using a compiler other than IBM COBOL II</td></tr> </table>	Code	Program Language	A	Assembler	C	C language (IBM C/370 Compiler only)	K	COBOL programs compiled using the IBM COBOL II compiler	J	COBOL programs compiled using a compiler other than IBM COBOL II
Code	Program Language										
A	Assembler										
C	C language (IBM C/370 Compiler only)										
K	COBOL programs compiled using the IBM COBOL II compiler										
J	COBOL programs compiled using a compiler other than IBM COBOL II										
Field Exit UE	Indicates whether the program is a Field Exit user exit program. Enter Y if it is or N if it is not.										
Post-Translate UE	Indicates whether the program is a Post-Translate user exit program. Enter Y if it is or N if it is not.										
Pre-Translate UE	Indicates whether the program is a Pre-Translate user exit program. Enter Y if it is or N if it is not.										
Encryption UE	Indicates whether the program is an Encryption user exit program. Enter Y if it is or N if it is not.										
Authentication UE	Indicates whether the program is an Authentication user exit program. Enter Y if it is or N if it is not.										
Compression UE	Indicates whether the program is a Compression user exit program. Enter Y if it is or N if it is not.										
Filtering UE	Indicates whether the program is a Filtering user exit program. Enter Y if it is or N if it is not.										
Monitor UE	Indicates whether the program is a Monitor user exit program. Enter Y if it is or N if it is not.										
Communication UE	Indicates whether the program is a Communication user exit program. Enter Y if it is or N if it is not.										
Message Process UE	Indicates whether the program is a Message Process user exit program. Enter Y if it is or N if it is not.										
Point-to-point UE	Indicates whether the program is a Point-to-point user exit program. Enter Y if it is or N if it is not.										
Envelope UE	Indicates whether the program is an Envelope user exit program. Enter Y if it is or N if it is not.										

MQSeries Queue Profile (MQSERIES)

The MQSeries Queue profile (MQSERIES) is used to associate logical names with real MQSeries Queues. The profile also provides a way to associate processing options DataInterchange will use whenever the MQSeries Queue profile member is supplied. You must also coordinate with your MQSeries administrator the creation and proper authorization of the MQSeries Queues you intend to use with DataInterchange. Once your MQSeries administrator has created the appropriate MQSeries Queues, you would then create DataInterchange MQSeries Queue profile members on behalf of these real MQSeries Queues. Within DataInterchange you will *always* refer to MQSeries Queues via their DataInterchange profile member names, not the real MQSeries names.

The following panel shows the definition of the MQSeries Queue profile (MQSERIES).

Defining Operational Profiles

```
PM05                                View Profile Definition                1 to 6 of 6

Profile ID: MQSERIES  Profile description: Application Definition Profile

Key==>  Field Label      Length Type Description
Queue Profile ID 008 CH Profile member identifier
Full Queue Name 048 CH MQSeries Queue Name
Queue Manager Name 048 CH MQSeries Queue Manager
Destructive Reads 001 CH Destructive Reads (Y,N)
Syncpoint Control 001 CH Syncpoint Control (Y,N)
Maximum Msg Length 008 CH Maximum message length
```

Field	Description
Queue Profile ID	The name of the member. This field relates a logical name to a real MQSeries Queue name and the properties DataInterchange should use when accessing the queue.
Full Queue Name	The real name of the MQSeries Queue. During MQSeries administration, queues can be defined and their full names are supplied at that time. This field must match <i>exactly</i> with the name given to MQSeries. Queue names are case sensitive, so care must be taken when they are entered into DataInterchange. This field is mandatory.
Queue Manager Name	The name of the MQSeries Queue Manager the queue is associated with. This field must match <i>exactly</i> with the MQSeries Queue Manager name since the name is case sensitive within MQSeries. This field is optional and if not supplied, the default manager will be used.
Destructive Reads	Indicates whether or not the reading from the queue should cause the message to be deleted by MQSeries. This field is optional and if not supplied, the default value is Y. If N is specified all reads from the queue will be non-destructive and browse mode will be used.
Syncpoint Control	Indicates whether or not operations against this queue should be under syncpoint control. If syncpoint control is used, then messages are either committed or backed out as a group from the previous commit or task initiation. If syncpoint control is not used, each message is an independent entity. The default value is Y, syncpoint control is active. A value of N indicates syncpoint control is not to be used on this queue.
Maximum Msg Length	The maximum length of messages associated with this queue if the value is greater than 32700. DataInterchange must allocate data storage buffers for messages written to and read from MQSeries queues. The default size of these buffers is 32700. You can increase (or decrease) the maximum message length by entering a value in this field. For example, if you know this queue might have message sizes as large as 1 MB, then you must enter at least 1048576. Conversely, if you know the maximum message size will be 80, you may optionally enter this value to save storage during DataInterchange processing.

Security Profile (SECUPROF)

The security profile contains information used in the following processes:

Process	Description
Encryption	Protects data against unauthorized viewing
Authentication	Protects data against unauthorized changes
Filtering	Verifies that data does not contain characters with special meaning to the network
Compression	Shortens the length of data for more efficient transmission and storage

You can provide members to handle either encryption or authentication, or to handle both. Filtering and compression work only in conjunction with encryption. Neither filtering nor compression works as a separate process. You can associate the same security member with more than one trading partner by entering the member name in the **Security ID** field of the trading partner profile.

You can add as many members as you need to define the encryption and authentication processes you and your trading partners have agreed to use. DataInterchange supplies a member named IBMNSP. This member identifies programs that request services from the IBM 4753 Network Security Processor MVS Program.

For details about the programs identified in the profile, see the *DataInterchange Programmer's Reference*.

The following panel shows the definition of the security profile (SECUPROF).

PM05	View Profile Definition			1 to 12 of 12
Profile ID: SECUPROF		Profile description: Security profile		
Key==>	Field Label	Length	Type	Description
	Security ID	008	CH	Security identification
	Originator name	016	CH	Originator name
	Recipient name	016	CH	Recipient name
	Auth. type	001	CH	Authentication type (0,1)
	Auth. code	001	CH	Authentication code
	Encr. type	001	CH	Encryption type (0,1,2)
	Filtering type	001	CH	Filtering type (0,1,2,3,4)
	Encr. program	008	CH	Encryption program
	Auth. program	008	CH	Authentication program
	Comp. program	008	CH	Compression program
	Filtering program	008	CH	Filtering program
	Buffer size	005	CH	Buffer size for programs

Field	Description
Security ID	Indicates the name that you and DataInterchange use to refer to this member. Enter this same name in the trading partner profile member to identify the data protection and compression processes that apply when receiving from a trading partner. Enter it in the trading partner usage for sending to identify the processes that apply when sending to that trading partner.
Originator name	Indicates the name of the process that performs encryption or authentication of data to be sent, or that originates a cryptographic service message. This name is used when building the security segments for sending.
Recipient name	Indicates the name of the process that performs decryption or authentication of received data, or is the destination of a cryptographic service message. This name is used when building the security segments for sending.

Defining Operational Profiles

Field	Description												
Auth. type	Indicates whether authentication is needed. <ul style="list-style-type: none">• 1 indicates authentication.• 0 indicates no authentication.												
Auth. code	Indicates the authentication option. <ul style="list-style-type: none">• 1 indicates binary data.• 2 indicates coded character data.												
Encr. type	Indicates the encryption option used with the filtering type. Valid values are: <table><tr><th>Value</th><th>Description</th></tr><tr><td>0</td><td>No encryption</td></tr><tr><td>1</td><td>Cipher block chaining</td></tr><tr><td>2</td><td>Cipher feedback</td></tr></table>	Value	Description	0	No encryption	1	Cipher block chaining	2	Cipher feedback				
Value	Description												
0	No encryption												
1	Cipher block chaining												
2	Cipher feedback												
Filtering type	Indicates the filtering option used with the encryption type. Valid values are: <table><tr><th>Value</th><th>Description</th></tr><tr><td>0</td><td>No filter</td></tr><tr><td>1</td><td>Hexadecimal filter</td></tr><tr><td>2</td><td>ASCII filter</td></tr><tr><td>3</td><td>ASCII/BAUDOT filter</td></tr><tr><td>4</td><td>User-defined filter</td></tr></table>	Value	Description	0	No filter	1	Hexadecimal filter	2	ASCII filter	3	ASCII/BAUDOT filter	4	User-defined filter
Value	Description												
0	No filter												
1	Hexadecimal filter												
2	ASCII filter												
3	ASCII/BAUDOT filter												
4	User-defined filter												
Encr. program	Indicates the name of your encryption program. <ul style="list-style-type: none">• For sending, the enveloper calls this program if the trading partner usage specifies group or transaction encryption.• For receiving, the deenveloper calls this program if group or transaction security segments are present and indicate that decryption is required. The trading partner profile indicates which security member is used. <p>The user program information profile (ADAMCTL) must contain an entry for the encryption program.</p> <p>DataInterchange provides an encryption interface program named IBMNSPE (load module EDITREE). It does not require an entry in the ADAMCTL profile.</p>												
Auth. program	Indicates the name of your authentication program. <ul style="list-style-type: none">• For sending, the enveloper calls this program if the trading partner usage specifies group or transaction authentication.• For receiving, the deenveloper calls this program if the group or transaction security segments are present and indicate that authentication is required. The trading partner profile indicates which security member is used. <p>The user program information profile (ADAMCTL) must have an entry for this program.</p> <p>DataInterchange provides an authentication program named IBMNSPA (load module EDITRAA). IBMNSPA does not require an entry in the ADAMCTL profile.</p>												

Field	Description										
Comp. program	<p>Indicates the name of your compression program. You must also specify an encryption program in the Encryption program field.</p> <ul style="list-style-type: none"> For sending, the enveloper calls this program if the trading partner usage or trading partner profile specifies the name of this security member. For receiving, the deenveloper calls this program to decompress a transaction if group or transaction security segments are present and indicate that decryption is required. The trading partner profile specifies the name of this security member. <p>The user program information profile (ADAMCTL) must have an entry for this program.</p>										
Filtering program	<p>Indicates the name of your filtering program. You must also specify an encryption program in the Encryption program field.</p> <ul style="list-style-type: none"> For sending, the enveloper calls this program if the trading partner usage or trading partner profile specifies the name of this security member. For receiving, the deenveloper calls this program if group or transaction security segments are present and indicate that decryption and filtering are required. The trading partner profile specifies the name of this security member. <p>The user program information profile (ADAMCTL) must have an entry for this program.</p> <p>DataInterchange provides the following filtering routines:</p> <table> <tr> <th>Routine</th><th>Description</th></tr> <tr> <td>EDIHEX</td><td>Hexadecimal filtering</td></tr> <tr> <td>EDIASCII</td><td>ASCII filtering</td></tr> <tr> <td>EDIBAUDO</td><td>ASCII/BAUDOT filtering</td></tr> <tr> <td>IBMFILTR</td><td>Calls one of the above routines depending on the filtering type requested</td></tr> </table> <p>These routines do not require entries in the ADAMCTL profile. For a sample listing of each routine, see the <i>DataInterchange Programmer's Reference</i>.</p>	Routine	Description	EDIHEX	Hexadecimal filtering	EDIASCII	ASCII filtering	EDIBAUDO	ASCII/BAUDOT filtering	IBMFILTR	Calls one of the above routines depending on the filtering type requested
Routine	Description										
EDIHEX	Hexadecimal filtering										
EDIASCII	ASCII filtering										
EDIBAUDO	ASCII/BAUDOT filtering										
IBMFILTR	Calls one of the above routines depending on the filtering type requested										
Buffer size	<p>Indicates the length, in bytes, of the buffer used by the encryption, authentication, compression, or filtering program. The buffer size determines how much data is passed to the encryption, authentication, compression, and filtering programs at one time.</p> <p>The interface provides routines for handling data that is longer than the buffer size. For more information, see the <i>DataInterchange Programmer's Reference</i>.</p>										

Application Definition Profile (APPDEFS)

The application definition profile (APPDEFS) is used with the APPLID value (EDIMP for online administration) to establish settings at an application level, as opposed to system wide settings. Each APPDEFS profile member is associated with an application ID.

The following panel shows the definition of the application definition profile (APPDEFS).

Defining Operational Profiles

PM05 View Profile Definition 1 to 13 of 13

Profile ID: APPDEFS Profile description: Application Definition Profile

Key==>	Field Label	Length	Type	Description
	Application ID	008	CH	Application ID
	ACTLOGS member	008	CH	ACTLOGS member name
	Mgt. Rpt. active?	001	CH	Management reporting active (Y,N)
	Trx. Store active?	001	CH	Transaction Store active (Y,N,S,E)
	Trx. Image wanted?	001	CH	Transaction Image wanted (Y,N,S,E)
	Monitor Program	008	CH	Monitor user exit name
	Use prod usage?	001	CH	Use production usage (Y,N)
	Log standard data?	001	CH	Log standard data (Y,N,)
	FA image wanted?	001	CH	Funct. Ack Image Wanted (Y,N,S,E)
	Event log active?	001	CH	Event log active (Y,N)
	ALPHANUM table	008	CH	Alphanumeric data validation table
	CHARSET table	008	CH	Character data validation table
	Century ctrl year	002	CH	Century control year

Field

Application ID

Description

Indicates the name of the member. This field relates to the APPLID value supplied during initiation of the DataInterchange Utility or the initialization API request. See the *DataInterchange Programmer's Reference* for more information on APPLID. If you start DataInterchange with an APPLID that does not have a matching APPDEFS profile member, the APPLID will be used as the ACTLOGS member and management reporting will be made active. You can switch APPLIDs dynamically by using DataInterchange's application program interface or user exit facility.

ACTLOGS member

Indicates the name of the activity log profile member to be used with this application. This is an indirect way to associate an event log with an application. If a member name is not supplied, the **Application ID** field is used as the ACTLOGS member.

Mgt. Rpt. active?

Indicates whether management reporting statistics should be gathered for the application. Valid values are Y (yes) or N (no). The default is Y.

Trx. Store active?

Indicates whether the Transaction Store Database should be used. Any value other than 'N' will indicate that the Store should be active. The default is active (Y).

Value

Description

Y	Store is always active
N	Store is never active
S	Store is only active for transactions that translate successfully
E	Store is only active for transactions that do not translate successfully

Trx. Image wanted?

Indicates whether the Transaction Image should be saved. Any value other than N will indicate that the Image should be saved. The default is save the image (Y).

Value

Description

Y	Images are always written to the store
N	Images are never written to the store
S	Images are written to the store only for transactions that translate successfully
E	Images are written to the store only for transactions that do not translate successfully

Field	Description										
Monitor Program	This is a DataInterchange for CICS field only. Enter the name of the user exit program that DataInterchange will link to during enveloping and deenveloping. This user exit will be linked once a complete envelope is generated or deenveloped. If more than one envelope is generated or deenveloped, the user exit will be invoked for each envelope. The user exit (working in conjunction with Expedite/CICS) can be used to provide EDI performance statistics. For more information, see the <i>DataInterchange Programmer's Reference</i> .										
Use Prod Usage?	<p>This controls if it is acceptable for a production usage to be used for test messages when an active test usage does not exist or for information messages when an active information usage does not exist.</p> <p>The usual mode of operation for a test message is that a test usage will be used if an active test usage exists. Otherwise, a production usage will be used.</p> <p>The usual mode of operation for an information message is that an information usage will be used if an active information usage exists. Otherwise, a production usage will be used.</p> <p>A value of N will override the usual mode of operation and will disallow the use of production usages for both test and information messages.</p> <table> <tr> <th>Value</th><th>Description</th></tr> <tr> <td>Y</td><td>Allowed to use production usages for test and information messages.</td></tr> <tr> <td>N</td><td>Production usage not allowed for test and information messages.</td></tr> </table>	Value	Description	Y	Allowed to use production usages for test and information messages.	N	Production usage not allowed for test and information messages.				
Value	Description										
Y	Allowed to use production usages for test and information messages.										
N	Production usage not allowed for test and information messages.										
Log standard data?	Enter a Y (for yes) to always log the standard data during a translation. Enter an N (for no) to bypass the logging of the standard data. Any other value will cause the translator to use the setting of the Log standard data field in the Trading Partner profile (TPPROF) to determine if the standard data should be logged.										
FA image wanted?	<p>Indicates whether the functional acknowledgment image should be saved. The default is (Y), to save the image.</p> <table> <tr> <th>Value</th><th>Description</th></tr> <tr> <td>Y</td><td>FA images are always written to the store</td></tr> <tr> <td>N</td><td>FA images are never written to the store</td></tr> <tr> <td>S</td><td>FA images are written to the store only for inbound transactions that translate successfully</td></tr> <tr> <td>E</td><td>FA images are written to the store only for inbound transactions that fail standard compliance checks</td></tr> </table> <p>Whether or not a record of the functional acknowledgment is written to the store depends on the Trx. Store Active? flag. Additionally, only outbound functional acknowledgments generated by DataInterchange are affected by this flag. This does not control the writing of the inbound acknowledgments. Inbound acknowledgments are handled like any other EDI transactions.</p> <p>Attention: DataInterchange reprocessing commands RECONSTRUCT, REENVELOPE, and RETRANSLATE may not generate the same results as the original process.</p>	Value	Description	Y	FA images are always written to the store	N	FA images are never written to the store	S	FA images are written to the store only for inbound transactions that translate successfully	E	FA images are written to the store only for inbound transactions that fail standard compliance checks
Value	Description										
Y	FA images are always written to the store										
N	FA images are never written to the store										
S	FA images are written to the store only for inbound transactions that translate successfully										
E	FA images are written to the store only for inbound transactions that fail standard compliance checks										
Event log active?	This indicator allows you turn off event logging. The default value is Y. With a value of N, the event log is turned off and there is no attempt by DataInterchange to access it either by writing messages to it or by reading its records for display.										

Defining Operational Profiles

Field	Description
ALPHANUM table	The name of the alphanumeric data validation table used instead of the default. This table is used for administrative panel edits and translation processes. The default is ALPHANUM.
CHARSET table	The name of the multiple character data validation table used instead of the default. The default is CHARSET.
Century ctrl year	Indicates the 2-digit value that DataInterchange uses to determine the century for a 2-digit year field. For years greater than this value, generated century will be 19. For years less than this value, century will be 20. The default century control year is 10. For example, if you set the value at 50, then all 2-digit years 1-49 are assumed to be in the 1900s.
	Note: The trading partner mapping keyword DICCCTRL overrides this value.

System Profile (SYSPROF)

Currently, the system profile is used only for DataInterchange for CICS. Each system profile member identifies a CICS region in which DataInterchange for CICS is used. It is used to indicate whether or not the DataInterchange persistent environment is to be used, and provides some information related to the persistent environment. If you are not running CICS/ESA or you do not define a system profile member for your CICS region running DataInterchange for CICS, the persistent environment will not become active. You must define a system profile member for each CICS region that will use the DataInterchange for CICS persistent environment. Refer to “Using DataInterchange in the CICS Environment” in the *DataInterchange Programmer's Reference* for more information on the persistent environment.

```
PM10                                Update Profile Member          1 to 3 of 3

Profile ID . . . : SYSPROF

System ID . . . . : ZBA5EDI1
Persist. active?   Y
Persist. size . . . 0064
Persist. threads   04
```

Field	Description
System ID	Identifies the APPLID of the CICS/ESA region running DataInterchange for CICS. The APPLID can be determined by entering CEMT INQUIRE TASK from a native CICS screen and looking at the bottom right corner of the displayed result.
Persist. active?	Indicates whether or not you want to initialize DataInterchange for CICS's persistent environment. Enter Y (yes) or N (no). If you change this value and wish the change to be reflected immediately in DataInterchange for CICS, you must execute CICS transaction EDIT. EDIT causes the DataInterchange for CICS session to terminate. When it is reinitialized, the persistent environment will or will not be initialized, based on the value in this field.

Field	Description
Persist. size	Enter the maximum size of 4 K blocks that may be allocated to the data space used by the DataInterchange for CICS persistent environment. The default value is 239 (239 x 4 K = 978,944 bytes). The actual size of the persistent environment may be less than the maximum size specified, but it will never be larger. The valid range is from 64 to 9997. However, it may not exceed the installation defined data space maximum value. The default maximum value is 239. Your installation may use the installation exit IEFUSI to change the default. Your systems programmer may advise you further on this value.
Persist. threads	Enter the number of MVS subtasks that will be created to manage the persistent environment. Valid values are from 1 to 16. The default is 4. Higher numbers allow more concurrent processing, but this is only useful in installations where much DataInterchange for CICS concurrent processing is expected. This value should not exceed the number of DataInterchange threads that may execute concurrently on the CICS region.

Chapter 5. Establishing Communications with Trading Partners

Overview of Trading Partner Communications	5-1
Overview of the Minimal Trading Partners Feature	5-2
Setting Up the Network Profile	5-4
Connecting to GEIS	5-4
Connecting to IBM Global Network	5-5
Updating a Network Profile Member	5-5
Connecting to Other Networks	5-8
Adding a Network Profile Member	5-8
Using MQSeries Queues	5-13
Setting Up the Requestor Profile	5-14
Adding a Requestor Profile Member	5-14
Setting Up the Trading Partner Profile	5-23
Adding a Trading Partner Profile Member	5-23
Utilizing Control Numbers	5-39
Assigning Control Numbers by Standard Transaction ID	5-40
Group Control Number Masks	5-40
Trading Partner Determination	5-42
Handling a Trading Partner in Multiple Environments	5-42
Continuous Receive Facility	5-43
Setting Up the Continuous Receive Profile	5-44
Requesting, Sending, and Receiving Functional Acknowledgments	5-48
Requesting Functional Acknowledgments	5-48
Sending Functional Acknowledgments	5-48
Receiving Functional Acknowledgments	5-49
Delayed Functional Acknowledgments	5-50
Interchange Acknowledgments (TA1 Segment)	5-50

Chapter 5. Establishing Communications with Trading Partners

This chapter provides information about defining your network, yourself, and your trading partners to DataInterchange. These tasks are necessary to exchange data with your trading partners.

Overview of Trading Partner Communications

To exchange data with your trading partners, DataInterchange needs information about the network, your organization, and your trading partners. DataInterchange then uses its communications routine to send the data to the network. Your trading partners retrieve the data from the network to their system.

To set up communications with your trading partners, you must do the following:

- | | |
|--|--|
| 1. Set up the MQSeries Queue profile. | This profile defines logical names with real MQSeries Queues. The profile also provides a way to associate processing options DataInterchange will use whenever the MQSeries Queue profile member is supplied. |
| 2. Set up the network profile. | This profile defines which networks you use to exchange information with your trading partners, and which communication routine you use. |
| 3. Set up the Requestor profile. | This profile defines the individuals or group of individuals, such as an accounting department, who use DataInterchange. You can define as many individuals or groups as needed by your organization. |
| 4. Set up the Trading Partner profile. | This profile defines your application trading partners and EDI trading partners, and includes information about account numbers or account IDs, user IDs, who pays for network charges, and so forth. |

You can eliminate the network and go directly from a DataInterchange system to another EDI system using a point-to-point setup. For more information about this type of setup, see the chapter on interfacing with other networks and applications in *DataInterchange Programmer's Reference*.

DataInterchange also supports exchanging data with your trading partner using an MQSeries message queue. You must define an MQSeries profile number for each message queue. For detailed information on setting up MQSeries communications, see "Interfacing DataInterchange with MQSeries" on page 6-36 in the *DataInterchange Programmer's Guide*.

CICS systems can establish a process that automates receiving transactions. See the *DataInterchange Programmer's Reference* and "Continuous Receive Facility" on page 5-43 for details.

DataInterchange uses *profiles* to store the information it needs to communicate with your trading partners. Although DataInterchange is shipped with many profiles, this chapter only describes the profiles needed to establish trading partner communications.

DataInterchange provides the profiles. It also provides several members for some profiles that you can use without changing. For other profiles, you must either add profile members or update those that are

supplied. You must have the authority to work with the profiles mentioned above (see Appendix A, "Security").

Overview of the Minimal Trading Partners Feature

The Minimal Trading Partners and Usages feature of DataInterchange allows users with large numbers of trading partners to minimize the number of trading partner profiles and usages required for a given set of trading partners and maps. To fully exploit the feature, it is necessary for the customer to take responsibility for determining the interchange ID and qualifier values that will appear in the interchange header segment. They must pass these values via the C record if C and D records are being used, or within their application data if raw data is being used. Since the interchange IDs and qualifiers are being provided by the application, it is not necessary to create trading partner profiles for your trading partners in DataInterchange.

If you have trading partners that require special processing, then you must identify them by creating a trading partner profile for them. Therefore, you may still enter some trading partners even if you are using Minimal Trading Partner and Usages. For example, if company A has 10,000 trading partners, and one of those partners (company B) requires a different set of delimiters than all the rest, then company A must create a trading partner profile for company B, specifying the delimiters to be used. Company A would not have to create (and maintain) trading partner profiles for the other 9,999 trading partners.

As the preceding example suggests, the existence of a trading partner profile for company B will override the set of delimiters used when sending to company B. How does company A specify which delimiters are to be used when it sends to the other 9,999 trading partners? The answer is that it must create a trading partner profile for company A. This trading partner profile will become the default trading partner profile (including delimiters) to be used whenever company A sends a message or transaction to a trading partner not defined in the trading partner profile. Company A's trading partner profile will also be the default trading partner profile used whenever company A receives a message or transaction from a trading partner not defined in the trading partner profile.

Though company A and B are both entered in the trading partner profile, they are not the same kind of trading partner. In fact, they are complementary. On an outbound translation (PERFORM TRANSLATE AND ENVELOPE), company A is the sender and company B is the receiver. On an inbound translation (PERFORM DEENVELOPE AND TRANSLATE), company B is the sender and company A is the receiver. Trading partners are differentiated within the trading partner profile using the Trading Partner Type field. A trading partner can be designated to be an application trading partner, an EDI trading partner, or both. Entities that were known formerly as simply "trading partners" are now considered "EDI trading partners." They are called EDI trading partners because they send EDI data to the translator to be translated. Application Trading Partners send application data to the translator to be translated. A trading partner designated as "both" may send either EDI or application data to the translator.

The fundamental difference between application and EDI trading partners is that the former deal exclusively with application data — which can include either data that comes from a specific entity, such as a division or department, or data of a certain document type, such as purchase orders, invoices, etc. — while the latter send and receive EDI data.

Application trading partners can also be conceived of as hubs, with EDI trading partners being the spokes that radiate from these hubs. In this arrangement, a hub can be an entity within a company (for example, a division or department such as Purchasing or Warehousing) or an entity that is external to it (for example, another company).

In addition to being either EDI- or Application-oriented, trading partners within DataInterchange are considered to be either KNOWN or UNKNOWN. A KNOWN trading partner is one for which there exists a

trading partner profile. In the previous example, when company B was entered into the trading partner profile, company B became KNOWN to DataInterchange. Company A was also entered into the trading partner profile and therefore become KNOWN to DataInterchange. The other 9,999 trading partners were UNKNOWN to DataInterchange. Within DataInterchange, we refer to the set of all trading partners, whether KNOWN or UNKNOWN, as ANY trading partner.

Usages relate trading partners to maps by telling DataInterchange to “use this map when this application trading partner sends a purchase order (or another type of transaction) to this EDI trading partner.” At the most fundamental level, a usage is the combination of a map name, an application trading partner name, and an EDI trading partner name. In addition to these fundamental data items, usages also allow you to specify a set of translation options that apply to the relationship. An example of a translation option would be Acceptable Error Level, which will affect whether the translator accepts or rejects a given message.

When specifying the application and EDI trading partners, you have two special values in both send and receive usages, KNOWN and ANY. As discussed previously, a KNOWN trading partner is any trading partner in the trading partner profile. ANY refers to any trading partner, whether it can be found in the trading partner profile or not. This allows DataInterchange to distinguish between a “specific trading partner,” where a trading partner nickname is specified, and a “generic trading partner,” indicated by use of the KNOWN and ANY keywords. The trading partner nickname is the key to distinguishing between specific and generic trading partners.

Continuing the preceding example of company A and its 10,000 trading partners, how many usages will the DataInterchange administrator have to create to link those trading partners to a single map? Previously, we discussed the fact that company B required a different set of delimiters than the rest of the trading partners. This required that we enter company B into the trading partner profile and specify the desired delimiters there. So far, we have not discussed any difference in the translation options to be used for any of the 10,000 trading partners, so only one usage would be required. In English, the usage relationship required would be read: “When company A sends this type of data to ANY trading partner (either the KNOWN trading partner company B, which is in the trading partner profile, or the 9,999 UNKNOWN trading partners that are not in the trading partner profile), use this map to translate the data.” You would code this in the send usage by specifying company A as the sending trading partner and specifying the keyword ANY for the receiving trading partner. This type of usage is known in DataInterchange as a specific-generic usage, because one of the trading partners is specific (company A), while the other is generic (the keyword ANY).

But what if company C requires a different Acceptable Error Level from the other 9,999 trading partners? To be able to refer to company C, we must add company C to the trading partner profile, even though none of its trading profile partner options are different from the default options. Finally, we must add a usage and specify the Acceptable Error Level to be used when company A sends to company C. You would code this in the send usage by specifying company A as the sending trading partner and company C for the receiving trading partner. This type of usage is known in DataInterchange as specific-specific usage, because both trading partners are specific (company A and company B). There are actually now two usages that specify two different sets of translation options that could be used when company A sends to company C: the company A to anybody usage and the company A to company C usage. To determine which usage to use, the translator arranges all candidate usages into a hierarchy and uses the one most specific to the current situation, in this case, company A to company C.

The hierarchy used by the translator has four general classes of combinations, each of which is described below. They are listed in order of precedence; in other words, the first one listed (specific-specific) will always be the one used if it is present.

- *Specific application and EDI trading partners.* DataInterchange translates this as meaning, “use this map when this application trading partner trades with this EDI trading partner.”

Communications with Trading Partners

- *Specific application trading partner, generic EDI trading partner.* DataInterchange translates this as meaning, “use this map whenever this application trading partner trades with any KNOWN or ANY trading partner as indicated by the keyword in the EDI trading partner field.”
- *Generic application trading partner, specific EDI trading partner.* DataInterchange translates this as meaning, “use this map whenever any KNOWN or ANY application trading partner (as indicated by the keyword in the application trading partner field) trades with this specific EDI trading partner.”
- *Generic application trading partner, generic EDI trading partner.* DataInterchange translates this as meaning, “use this map whenever any KNOWN or ANY application trading partner (as indicated by the keyword in the application trading partner field) trades with any KNOWN or ANY EDI trading partner (as indicated by the keyword in the EDI trading partner field.”

Setting Up the Network Profile

The *network profile* contains members that describe the networks you can use to exchange data with trading partners. DataInterchange provides members for the IBM Global Network (IGN) and the General Electric Information Services (GEIS) Company network. Table 5-1 describes network profile members provided by DataInterchange.

Table 5-1. Network Profile Members

Member	Communication Routine	Network Program	Used With:
GEIS	GEISVAN	DSXMIT2	General Electric Information Services Company
IINB1	VANIINB1	IEBASE	Expedite Base/MVS Version 1 Release 1
IINB41	VANIINB1	IEBASE	Expedite Base/MVS Version 4 Release 1
IINB42	VANIINB1	IEBASE	Expedite Base/MVS Version 4 Release 2 and higher
IINCICS	VANINFC	EXPOICMD	Expedite/CICS
IINR3	VANIINR3	TPMAIN	Expedite/MVS Version 1 Release 3
MQSAMP	VANIMQ	EDIMQSR	Sample MQSeries Network
TIGD31	TIGCMD31	EDIBTCH	InTouch* Gateway Release 3.1
TIGV31	TIGCMV31	EDIBTCH	InTouch* Gateway Release 3.1

Note: Before you can use IINB41, Information Exchange 4.1 services must be available for your location.

If you are not using the Global Network, the General Electric Information Services (GEIS) network, or the InTouch* Gateway, or if you want to use a point-to-point connection that bypasses the mailbox or use MQSeries queues to send and receive data, you must add a member to the network profile. See “Connecting to Other Networks” on page 5-8 for information about other networks. For more information about exchanging data with your trading partner using MQSeries message queues, see “Using MQSeries Queues” on page 5-13. For more information about using a point-to-point connection, writing your own CICS network program using the VANICICS communication routine provided with DataInterchange, or altering the operation of one or more network commands (NETOP section), see the *DataInterchange Programmer's Reference*.

Connecting to GEIS

If you are using the GEIS network, have the person who installed DataInterchange update the CLISTs EDIDB2 and EDIVSM to allocate the data sets needed by GEIS by changing the *GEIS (Y/N)* parameter of the CLISTs from N (the default) to Y.

Connecting to IBM Global Network

If you are using the IBM Global Network (IINB1, IINB41, IINB42, IINCICS, or IINR3), you should update the *Time zone* field in the network profile (see “Updating a Network Profile Member” on page 5-5). You probably will not need to change any other fields; however, if it is necessary to change a field for your site, see “Adding a Network Profile Member” on page 5-8 for a description of each field.

Note: If you are not updating an existing profile member or adding a new one, you can go directly to “Setting Up the Requestor Profile” on page 5-14.

Updating a Network Profile Member

To update a network profile member, follow these steps:

1. Log on to DataInterchange as described in “Accessing DataInterchange for MVS” on page 2-1 or “Accessing DataInterchange for CICS” on page 2-3.
2. From the Administrator’s Menu (MP01), select *Profiles*. The Profile Definitions panel (PM01) is displayed.
3. Type **M** in the action column next to NETPROF, and press Enter.

Log work with Members Print View			
PM01		Profile Definitions	1 to 17 of 17
Action	Profile ID	Profile Description	Log?
—	ACTLOGS	Activity log	N
—	ADAMCTL	User program information	N
—	APPDEFS	Application Definition Profile	N
—	CONTRECV	Continuous Receive Profile	N
—	E	EDIFACT standard envelope data	N
—	I	ICS standard envelope data	N
—	LANGPROF	Language profile	N
—	MQSERIES	MQSeries Queue Profile	N
—	NETOP	Network operation profile	N
m	NETPROF	Network profile	N
—	REQPROF	Requestor profile	N
—	SECUPROF	Security profile	N
—	SYSPROF	System profile	N
—	T	UN/TDI standard envelope data	N
—	TPPROF	Trading partner profile	Y
—	U	UCS standard envelope data	N
—	X	X12 standard envelope data	N

The Profile Members panel (PM07) is displayed.

4. Type **U** in the action column next to the network you are using, and press Enter. For example, to set up for IBM Global Network, type **U** next to IINB41, and press Enter.

Communications with Trading Partners

Add Copy Delete List Print Update View		
PM01 Profile Definitions		1 to 17 of 17
Action	Profile ID	PM07 Profile Members 1 to 10 of 10
-	ACTLOGS	Profile ID : NETPROF
-	ADAMCTL	Description: Network profile
-	APPDEFS	
-	CONTRECV	
-	E	Action Key
-	I	- GEIS
-	LANGPROF	- IINB1
-	MQSERIES	u IINB41
-	NETOP	- IINB42
m	NETPROF	- IINCICS
-	REQPROF	- IINR3
-	SECUPROF	- PTTOPT
-	MQSAMP	- TIGD31
-	SYSPROF	- TIGV31
-	T	
-	TPPROF	
-	U	
-	X	
X12 standard envelope data		N

The Update Profile Member panel (PM10) is displayed.

PM10	Update Profile Member	1 to 15 of 18
Profile ID . . . : NETPROF		
Network ID . . . : IINB41		
Network name . . . IBM Global Network_____		
Communication rtn VANIINB1		
Network program . . IEBASE__		
Network parameters_____		
Network input file INMSG__		
Input rec length 80__		
Trans data queue QDATA__		
Trans rec length 80__		
Time zone GMT__		
System type _____		
System level _____		
Msg text header . . T__		
Net output file . . OUTMSG__		
Message handler . . INB1MSG__		
Network sequence 00941		

PM10	Update Profile Member	15 to 18 of 18
Profile ID . . . : NETPROF		
Network ID . . . : IINB41		
Network sequence 00941		
Net acks file . . . _____		
Dial connect num _____		
Script name _____		

5. Complete the following fields:

In this field:

Time zone

Enter:

The code for your location's time zone:

Code	Description
AHD	Hawaii daylight time
AHS	Hawaii standard time
AST	Atlantic standard time
BST	British summer time
CDT	Central daylight time
CST	Central standard time
EAD	Eastern Australia time
EDT	Eastern daylight time
EMT	Eastern Mediterranean time
EST	Eastern standard time
GMT	Greenwich mean time
JST	Japan standard time
MDT	Mountain daylight time
MST	Mountain standard time
PDT	Pacific daylight time
PST	Pacific standard time
WED	Western Europe daylight time
YDT	Alaska daylight time
YST	Alaska standard time

You can also specify an offset from Greenwich mean time by indicating the number of hours and minutes east or west of the Greenwich meridian. The format is *dhhmm*, where *d* is E or W, *hh* indicates the hours, and *mm* indicates the minutes. For example, some valid values for IBM Global Network are:

Code	Description
E0000	Greenwich mean time
E0100	Western Europe standard time
E0900	Japan standard time
E1000	Eastern Australia daylight time
W0400	Eastern daylight time
W0500	Eastern standard time
W0500	Central daylight time
W0600	Central standard time
W0700	Mountain standard time
W0800	Pacific standard time

The default is E0000.

This field is required if you use IINR3, and if not provided, a communication error results.

IN reference: TIMEZONE.

Trans rec length

MVS users can indicate a maximum length of '9999'. CICS users can indicate up to '9999' explicitly, or, if using a temporary storage queue, they can use '0' or blank to indicate the maximum temporary storage queue length, which is 28000.

Communications with Trading Partners

6. Press Enter to save this information. DataInterchange updates the network profile, then redisplay the Profile Members panel (PM07).

7. Press F3 (Exit) twice to return to the Administrator's Menu.

Note: If you are not adding your own network, go to "Setting Up the Requestor Profile" on page 5-14.

Connecting to Other Networks

If you are connecting to a network other than IBM Global Network, GEIS, or the InTouch* Gateway, you must add a member describing the network to the network profile. See the *DataInterchange Programmer's Reference* for information about providing an interface to a network for which DataInterchange does not provide a network profile member.

Adding a Network Profile Member

To add a member to the network profile, follow these steps:

1. From the Administrator's Menu (MP01), select PROFILES. The Profile Definitions panel (PM01) is displayed.
2. Type **M** in the action column next to NETPROF, and press Enter.

Log work with Members Print View			
PM01		Profile Definitions	1 to 17 of 17
Action	Profile ID	Profile Description	Log?
—	ACTLOGS	Activity log	N
—	ADAMCTL	User program information	N
—	APPDEFS	Application Definition Profile	N
—	CONTRECV	Continuous Receive Profile	N
—	E	EDIFACT standard envelope data	N
—	I	ICS standard envelope data	N
—	LANGPROF	Language profile	N
—	MQSERIES	MQSeries Queue Profile	N
—	NETOP	Network operation profile	N
m	NETPROF	Network profile	N
—	REQPROF	Requestor profile	N
—	SECUPROF	Security profile	N
—	SYSPROF	System profile	N
—	T	UN/TDI standard envelope data	N
—	TPPROF	Trading partner profile	Y
—	U	UCS standard envelope data	N
—	X	X12 standard envelope data	N

The Profile Members panel (PM07) is displayed.

3. Type **A** in the action column next to any item, and press Enter.

Add Copy Delete List Print Update View			
PM01		Profile Definitions	1 to 17 of 17
Action	Profile ID	PM07 Profile Members 1 to 10 of 10	
-	ACTLOGS	Profile ID : NETPROF	
-	ADAMCTL	Description: Network profile	
-	APPDEFS	Action Key	
-	CONTRECV	a GEIS	
-	E	- IIN	
-	I	- IINB41	
-	LANGPROF	- IINB42	
-	MQSERIES	- IINCICS	
-	NETOP	- IINR3	
m	NETPROF	- IIND31	
-	REQPROF	- TIGV31	
-	SECUPROF	- PTTOPT	
-	SYSPROF		
-	T		
-	TPROFOF		
-	U		
-	X	X12 standard envelope data N	

The Add Profile Member panel (PM08) is displayed.

PM08 Add Profile Member		1 to 13 of 19
Profile ID: NETPROF		
Fill in the information below and press Enter to save this member. To stop entering members, press Exit or Cancel.		
Network ID	_____	
Network name . . .	_____	
Communication rtn	_____	
Network program . .	_____	
Network parameters	_____	
Network input file	_____	
Input rec length	_____	
Trans data queue	_____	
Trans rec length	_____	
Time zone	_____	
System type	_____	
System level . . .	_____	
Msg text header . .	_____	

PM08 Add Profile Member		13 to 19 of 19
Profile ID: NETPROF		
Fill in the information below and press Enter to save this member. To stop entering members, press Exit or Cancel.		
Msg text header . .	_____	
Net output file . .	_____	
Message handler . .	_____	
Network sequence	_____	
Net acks file . . .	_____	
Dial connect num	_____	
Script name	_____	

Communications with Trading Partners

4. Type the appropriate information in each field, and press Enter. If you need additional information about the use of a specific field, place the cursor in that field and press F1 (Help).

In this field:

Enter:

Network ID

A unique name to identify the network, such as MYNET or IINB41. This value is referenced by the requestor profile and the trading partner profile. Use the same ID throughout DataInterchange to refer to this network.

Network name

A descriptive name of the network, such as My EDI network. This field is optional.

Communication rtn

The name of the communication routine that builds network commands and invokes the network's send and receive program to process the commands. See Table 5-1 on page 5-4 for a list of the communication routines provided with DataInterchange. In addition, the PTTOPT communication routine for point-to-point connections is provided with DataInterchange.

Network program

The physical name of the send and receive program. This is the program invoked by the communication routine to process requests.

Network parameters

The parameters required by the network program. For example, for GEIS the parameter is the phone number for dial-up connection. These parameters are not used for point-to-point connections or for CICS. See the documentation provided by your vendor, or the programmer who provided the network program. The SENDMQ= keyword should be followed by the DataInterchange MQSeries Queue profile member name you will use to "send" data to. The RECEIVEMQ= keyword should be followed by the DataInterchange MQSeries Queue profile member name you will use to "receive" data from. Notice that the two parameters are blank space delimited; this must be adhered to. If you are performing one way communications with your trading partner, only the keyword-value combination of that direction is required.

Network input file

The data definition name (ddname) that contains the commands written for the network program to process. For IINB41, the name is INMSG. This field is not used for point-to-point connections or CICS.

Input rec length

The length of records in the network input file. For IBM Global Network, the record length is 80. For a point-to-point connection or CICS, leave this field blank.

In this field:

Trans data queue

Enter:

The ddname of the file that will hold the enveloped transactions waiting to be sent to your trading partner. This file is also used when queuing or sending transactions. If you leave this field blank, the file has one of the following names by default:

Name	Description
QDATA	For transactions enclosed in type X (ISA/IEA) or I (ICS/ICE) interchange envelopes, or for transactions without interchange envelopes.
QDATAE	For transactions enclosed in type E (UNB/UNZ) or T (STX/END) interchange envelopes.
QDATAU	For transactions enclosed in BG/EG interchange envelopes.

If the file contains type E (UNB/UNZ) or T (STX/END) envelopes, DataInterchange appends an E to the name; if the file contains BG/EG envelopes, it appends a U; and if the file contains type X (ISA/IEA) or I (ICS/ICE) envelopes, or transactions with no interchange envelope, it uses the name as supplied.

The type of send command (SENDX12, for example) determines which file is used. For example, if you enter the name SENDPO and use the file to send UNB/UNZ envelopes, DataInterchange expects to find an allocation for the ddname SENDPOE.

If you are using DI for CICS, enter the name of the temporary storage queue that will hold the enveloped transactions waiting to be sent to trading partners. If you leave this field blank, the temporary storage queue has one of the following names by default:

Name	Description
EDIQDAT	For transactions enclosed in type X (ISA/IEA) or I (ICS/ICE) interchange envelopes, or for transactions without interchange envelopes.
EDIQDATE	For transactions enclosed in type E (UNB/UNZ) or T (STX/END) interchange envelopes.
EDIQDATU	For transactions enclosed in type E (UNB/UNZ), or for transactions enclosed in type U (BG/EG) interchange envelopes.

If the temporary storage queue contains type E (UNB/UNZ) or T (STX/END) envelopes, DataInterchange appends an E to the name; if the temporary storage queue contains type U (BG/EG) envelopes, it appends a U; and if the queue contains type X (ISA/IEA) or I (ICS/ICE) envelopes, or transactions with no interchange envelope, it uses the name as supplied.

Note: You can override the transaction data queue ddname in the utility commands for sending transactions.

Communications with Trading Partners

In this field:

Trans rec length

Enter:

The length of records in the transaction data queue. The communication routine provided by DataInterchange (VANIINB41) ignores this field and uses the logical record length you allocated for the file. For DataInterchange for CICS, the maximum usable record length that can be used in a Temporary Storage Queue (TSQ) is 28000. To utilize all 28000 bytes in each record, a value of zero or blank should be entered in the TRANS REC LENGTH field. Otherwise, the maximum number that can be entered in this 4-character field is '9999'.

Time zone

The code for your location's time zone. The network specifies the allowable codes. See page 5-7 for the list of codes allowed by IBM Global Network.

System type

This field is optional. If used, the code describes the software you use to interface with IN. Valid values are:

Code	Description
------	-------------

01	Unknown system type
10	IBM expEDItE/PC
11	Expedite Base/2
12	Expedite Base/AIX
14	Expedite Base for SCO UNIX
15	Expedite Base/DOS
16	Expedite Base for SCO XENIX
17	Expedite Base for Windows
19	Expedite for Windows
20	IBM expEDItE/MVS Host
21	Expedite Base/MVS
22	TCP/IP FTP Gateway
30	IBM Mail Exchange
31	Expedite Base/VM
33	X.400 Gateway
40	Expedite/Direct
44	EDI VAN Interconnect
60	IBM MQSeries Services
61	IBM EDI Services
71	Expedite Base/400
80	Expedite/CICS
90	Information Exchange Administration Services
91	Expedite Async

System level

This field is optional. If used, it represents a value that indicates to your trading partner the level of your system, for example, B41. If you do not know what to type in this field, leave it blank.

Msg text header

The character that indicates the beginning of text for a message. For IBM Global Network, the character is T.

Net output file

The logical name of the file containing the network's responses to the command input file. For example, for IINB41 the name is OUTMSG. For a point-to-point connection, leave this field blank. This field is not used in DataInterchange for CICS.

In this field:	Enter:
Message handler	The name of the program that processes messages and network acknowledgments from the network. When you send a transaction, if your network returns a message to indicate the success or failure of your send request, the message handler program interprets the returned message and updates the status of the interchange. For example, for IINB1, IINB41, and IINB42, the name is INB1MSG; for IINR3, the name is INMSGHL; and for GEIS, the name is GEMSGHL. For a point-to-point connection, type NONE .
Network sequence	A number that DataInterchange increments and assigns to all outbound documents. You can set or reset the value at which sequential numbering begins.
Net acks file	The name of a file (MVS ddname) where you would like the network to write network acknowledgments when you request a status update. The network acknowledgments are read and evaluated by the message handler program. Note: This feature will be enabled in a future release of DataInterchange.
Dial connect num	The phone number to dial to connect to your network.
Script name	Optional. The name of a set of instructions that the communication software can use to process requests associated with this network profile member. The set of instructions would be part of the communication software package and not part of DataInterchange. See "Interfacing DataInterchange for CICS with SDM LinkPlus Interactive" in the <i>DataInterchange Programmer's Reference</i> . Note: You can override the script name in the utility commands for sending transactions.

5. Press Enter to save this information. DataInterchange updates the network profile, then redisplay the Profile Members panel (PM07).
6. Press F3 (Exit) twice to return to the Administrator's Menu.

Using MQSeries Queues

You can exchange data with your trading partners using MQSeries Queues. DataInterchange has been designed to prohibit the use of MQSeries Queues as the target of the envelope process, and they also cannot be used as input to the deenvelope process. But MQSeries Queues can be used as part of a logical network where enveloped data is sent to and received from MQSeries Queues. DataInterchange provides a sample network profile member named **MQSAMP** to assist you with the setup of this local network.

1. Have your MQSeries Administrator define the MQSeries Queue you plan on using.
2. Define DataInterchange MQSeries Queue profile members on behalf of the queues your MQSeries administrator has created.
3. Copy network profile member MQSAMP to a new network profile member. Here is the MQSAMP profile member:

```
PM11                                View Profile Member          1 to 19 of 19

Profile ID:  NETPROF

Network ID . . . . MQSAMP
Network name . . . Sample MQSeries Network
Communication rtn  VANIMQ
Network program . . EDIMQSR
Network parameters SENDMQ=mqprof1 RECEIVEMQ=mqprof2
Network input file
Input rec length
Trans data queue   QDATA
Trans rec length
Time zone . . . . . GMT
System type . . . .
System level . . . .
Msg text header . .
Net output file . .
Message handler . .
Network sequence . 000000
Net acks file . . .
Dial connect num .
Script name . . . .
```

4. Change the Network parameters field to match the DataInterchange MQSeries Queue profile member name(s) you created in step 2. The SENDMQ= keyword should be followed by the DataInterchange MQSeries Queue profile member name you will use to "send" data to. The RECEIVEMQ= keyword should be followed by the DataInterchange MQSeries Queue profile member name you will use to "receive" data from. Notice that the two parameters are blank space delimited; this must be adhered to. If you are performing one-way communications with your trading partner, only the keyword-value combination of that direction is required.
5. You may also update the Network name, Trans data queue, and Time zone fields. Please refer to their description in the section "Adding a Network Profile Member" on page 5-8.
6. Your new network profile member can be used within trading partner profiles and requestor profiles. Please refer to documentation on those profiles for more information.

Setting Up the Requestor Profile

The requestor profile contains members that describe the individual users or groups who request network services for sending or receiving transactions, messages, or files on your DataInterchange system. A requestor is similar to a network mailbox owner. You must identify each requestor by adding a member to the requestor profile. Although multiple requestors can use the same mailbox, you must still identify each requestor by adding a member to the requestor profile.

Adding a Requestor Profile Member

To add a member to the Requestor profile, follow these steps:

1. From the Administrator's Menu, select PROFILES. The Profile Definitions panel (PM01) is displayed.

2. Type **M** in the action column next to REQPROF, and press Enter.

Log work with Members Print View			
PM01		Profile Definitions	1 to 17 of 17
Action	Profile ID	Profile Description	Log?
-	ACTLOGS	Activity log	N
-	ADAMCTL	User program information	N
-	APPDEFS	Application Definition Profile	N
-	CONTRECV	Continuous Receive Profile	N
-	E	EDIFACT standard envelope data	N
-	I	ICS standard envelope data	N
-	LANGPROF	Language profile	N
-	MQSERIES	MQSeries Queue Profile	N
-	NETOP	Network operation profile	N
-	NETPROF	Network profile	N
m	REQPROF	Requestor profile	N
-	SECUPROF	Security profile	N
-	SYSPROF	System profile	N
-	T	UN/TDI standard envelope data	N
-	TPPROF	Trading partner profile	Y
-	U	UCS standard envelope data	N
-	X	X12 standard envelope data	N

The Profile Members panel (PM07) is displayed.

3. Type **A** in the action column next to any item, and press Enter.

Add Copy Delete List Print Update View																																																							
PM01		Profile Definitions	1 to 17 of 17																																																				
Action	Profile ID	<table> <tr> <td colspan="2">PM07</td><td>Profile Members</td><td>1 to 9 of 13</td></tr> <tr> <td colspan="4">Profile ID : REQPROF</td></tr> <tr> <td colspan="4">Description: Requestor profile</td></tr> <tr> <td>Action</td><td>Key</td><td colspan="2"></td></tr> <tr> <td>a</td><td>ACCTGROUP</td><td colspan="2"></td></tr> <tr> <td>-</td><td>AUDITGROUP</td><td colspan="2"></td></tr> <tr> <td>-</td><td>BAKERGROUP</td><td colspan="2"></td></tr> <tr> <td>-</td><td>DAVISTINA</td><td colspan="2"></td></tr> <tr> <td>-</td><td>INVENTORY</td><td colspan="2"></td></tr> <tr> <td>-</td><td>PAYROLL</td><td colspan="2"></td></tr> <tr> <td>-</td><td>PERSONNEL</td><td colspan="2"></td></tr> <tr> <td>-</td><td>WORKERA</td><td colspan="2"></td></tr> <tr> <td>-</td><td>WORKERB</td><td colspan="2"></td></tr> </table>		PM07		Profile Members	1 to 9 of 13	Profile ID : REQPROF				Description: Requestor profile				Action	Key			a	ACCTGROUP			-	AUDITGROUP			-	BAKERGROUP			-	DAVISTINA			-	INVENTORY			-	PAYROLL			-	PERSONNEL			-	WORKERA			-	WORKERB		
PM07		Profile Members	1 to 9 of 13																																																				
Profile ID : REQPROF																																																							
Description: Requestor profile																																																							
Action	Key																																																						
a	ACCTGROUP																																																						
-	AUDITGROUP																																																						
-	BAKERGROUP																																																						
-	DAVISTINA																																																						
-	INVENTORY																																																						
-	PAYROLL																																																						
-	PERSONNEL																																																						
-	WORKERA																																																						
-	WORKERB																																																						
-	U	UCS standard envelope data	N																																																				
-	X	X12 standard envelope data	N																																																				

The Add Profile Member panel (PM08) is displayed.

Communications with Trading Partners

PM08	Add Profile Member	1 to 13 of 20
Profile ID: REQPROF		
Fill in the information below and press Enter to save this member. To stop entering members, press Exit or Cancel.		
Requestor ID . . .	_____	
Network ID	_____	
Account number . .	_____	
User ID	_____	
Network password	_____	
Message user class	_____	
Receive file name	_____	
Network msg class	_____	
Network charges . .	_____	
Net acknowledgment	_____	
Destination verif	_____	
Retention period	_____	
EDI option	_____	

PM08	Add Profile Member	13 to 20 of 20
Profile ID: REQPROF		
Fill in the information below and press Enter to save this member. To stop entering members, press Exit or Cancel.		
EDI option	_____	
EDI proc override	_____	
Format override . .	_____	
Storage format . .	_____	
Network cmds file	_____	
Data line timeout	_____	
Remote status pgm	_____	
Alt net dial num	_____	
Compression	_____	
Priority	_____	

4. Type the appropriate information in each field, and press Enter. Use the F8 (Forward) function key to scroll to the second panel. If you need additional information about the use of a specific field, place the cursor in that field and press F1 (Help).

In this field:**Enter:**

Requestor ID

The name of the requestor who wants to use DataInterchange. A requestor ID often identifies a specific department within a company.

Network ID

The network that is used by this requestor. The name must match the name of a network in the network profile (NETPROF).

Account number

The account number or account ID assigned to the requestor by the network. The entry must be left-justified. For sending and receiving ISA/IEA type envelopes using IBM Global Network, the last position must be blank.

User ID

The user ID assigned to the requestor by the network. It must be left-justified.

The account number and user ID define your network mailbox, either the one you send data from or the one your trading partner sends data to.

In this field:

Network password

Enter:

The password required to access the mailbox identified in the ACCOUNT NUMBER and USER ID fields. For IBM Global Network, the first 8 characters are the current password, and the second 8 characters are the new password, used for changing your password. If you are not resetting your password, leave the second 8 characters blank.

IN reference: PASSWORD and NEWPSWRD.

Message user class

A code that you and your trading partner agree to use for identifying the class of information to be sent or received. Examples of classes are #E2, #EC, #EU, #EE, DEPT01, X12, MSG, FILE, EDI, or UCS. The message user class code allows you to select one type of information from a mailbox that can hold various types of information. The code can be omitted to indicate that all information for the mailbox is to be sent or received.

For EDIFACT and UNTDI messages, you can supply the message user class code in the UNB14 field of the E profile member (for EDIFACT) or the STX11 field of the T profile member (for UNTDI). For these types of envelopes, you should update the standard envelope data and assign data type AP to the application reference data element. An entry in the requestor profile field overrides the UNB14 and STX11 fields of the envelope profiles.

The Interactive Entry Facility receives the message user class from the document definition or from data entered into the SPECIFIC NAME field. The enveloper maps the value to the application reference data element if the data element data type is AP.

Tip: Do not receive standard and nonstandard data in the same receive request, because nonstandard files and free-form messages are not translated, and are treated as input errors. Use the message user class to receive standard and nonstandard data separately.

Receive file name

The data definition name (ddname) of the file into which data is written when receiving information from the network. The translator processes standard transactions from the named file. The *receive file name* must match the name in the EDI CLIST to receive transactions from the Interactive Entry Facility; DataInterchange supplies the name RECVDATA in the CLIST, by default. For applications requesting services from DataInterchange by the utility, the RECEIVE FILE NAME must match the ddname in your JCL.

You can override the receive file name in the utility commands for receiving transactions.

Note: For CICS, the receive file is a temporary storage queue. CICS applications should use an override in the receive command to avoid receiving EDI data for different applications into the same queue.

Network msg class

A code that indicates any special status of the data being sent. For IBM Global Network, T indicates test status and a blank indicates normal status.

This field is not used when receiving data. If you specify a trading partner when requesting network activity (send or receive), the value for this field is taken from the trading partner profile. Otherwise, the value for this field is taken from the requestor profile.

IIN reference: MSGNCLS.

IINB1, IINB41, and IINB42 reference: MODE

Communications with Trading Partners

In this field:

Network charges

Enter:

The charge code for network charges. This code allows the network to determine who should be charged when data is sent. Valid values are:

Code	Description
------	-------------

- | | |
|---|--|
| 1 | Receiver pays all charges. |
| 2 | Receiver pays all charges by agreement. If receiver has not agreed to pay all charges, charges are split between the sender and receiver. |
| 3 | Receiver pays all charges by agreement. If receiver has not agreed to pay all charges, charges are split between the sender and receiver, by agreement, or the sender pays all charges. Code 3 is the default for the Network charges field. |
| 4 | Charges are split between the sender and receiver by agreement, or the sender pays all charges. |
| 5 | Charges are split between the sender and receiver. |
| 6 | Sender pays all charges. |

This field is not used when receiving data. If you specify a trading partner when requesting network activity (send or receive), the value for this field is taken from the trading partner profile. Otherwise, the value for this field is taken from the requestor profile.

IINR3 reference: MSGCHRG.

IINB1, IINB41, and IINB42 reference: CHARGE.

For a detailed explanation of IBM Global Network message charges, see the *Understanding Information Exchange Message Charges*.

Net acknowledgment

The code that indicates which network acknowledgments, such as received by network (receipt), delivered to trading partner (delivery), or purge, that you want returned to you when sending data to trading partners. Valid values are:

Code	Description
------	-------------

- | | |
|---------|---|
| (blank) | No acknowledgments |
| R | Receipt acknowledgments only |
| D | Delivery acknowledgments only |
| B | Both receipt and delivery acknowledgments |
| A | Purge acknowledgments only |
| C | Both receipt and purge acknowledgments |
| E | Either purge or delivery acknowledgments |
| F | Receipt acknowledgments, and either delivery or purge acknowledgments |

This field is not used when receiving data. If you specify a trading partner when requesting network activity (send or receive), the value for this field is taken from the trading partner profile. Otherwise, the value for this field is taken from the requestor profile.

IINR3 reference: MSGRCPT.

IINB1, IINB41, and IINB42 reference: ACK.

In this field:

Destination verif

Enter:

The code that indicates whether the destination and authorization are to be verified before sending. Valid values are:

Code Description

N No verification (default)

Y Verification

F Verification and sending even if the destination is not verified (useful for intersystem addressing)

This field is not used when receiving data. If you specify a trading partner when requesting network activity (send or receive), the value for this field is taken from the trading partner profile. Otherwise, the value for this field is taken from the requestor profile.

IINR3 reference: MSGVCHK.

IINB1, IINB41, and IINB42 reference: VERIFY.

Retention period

The number of days that data is kept in a mailbox before it is purged. For IBM Global Network, enter blanks or 000 to use the default, or enter a range from 1 to 180.

This field is not used when receiving data. If you specify a trading partner when requesting network activity (send or receive), the value for this field is taken from the trading partner profile. Otherwise, the value for this field is taken from the requestor profile.

IINR3 reference: MSGRETN.

IINB1, IINB41, and IINB42 reference: RETAIN.

EDI option

Y if you want to store EDI segments as separate records, or **N** if you do not. The DataInterchange translator accepts data in either format.

This field is used when receiving or sending data. If you specify a trading partner when requesting network activity, the value for this field is taken from the trading partner profile. Otherwise, the value for this field is taken from the requestor profile.

IBM Global Network reference: EDIOPT.

EDI proc override

A code that indicates whether or not data you receive is to have special EDI processing, which consists of breaking the records by the segment delimiter. For IBM Global Network, valid values are:

Code Description

Y Perform EDI processing if the common data header indicates that the data in the file is in EDI standard format.

N Omit EDI processing, regardless of the common data header.

This field is not used when sending data. If you specify a trading partner when requesting network activity (send or receive), the value for this field is taken from the trading partner profile. Otherwise, the value for this field is taken from the requestor profile.

IINR3 reference: EDIPROC.

IINB1, IINB41, and IINB42 reference: AUTOEDI.

Communications with Trading Partners

In this field:

Format override

Enter:

Y if you want to use the storage format defined in the common data header, or **N** if you do not.

The common data header provides details (such as file name, carriage return, and line-feed options) that let the receiving interface reconstruct a received message into its original format. It also makes more information available to the recipient of a file. If there is no common data header, the format indicated in the STORAGE FORMAT field is used.

This field is not used when sending data. If you specify a trading partner when requesting network activity (send or receive), the value for this field is taken from the trading partner profile. Otherwise, the value for this field is taken from the requestor profile.

IINR3 reference: STGFORMO.

IINB1, IINB41, and IINB42 reference: DLMOVERRIDE.

Storage format

The code that tells the network how data is stored for free-form messages and files.

Consider the type of data you want to send and how the file is received when determining what values to select. Contact a representative from each network you are using for all available values.

For example, if you are using:

- Expedite/Base MVS Version 1.1 (IEBASE), valid values are:

Value	Description
-------	-------------

C	Stores each record with a carriage return and line-feed character and uses the end-of-file character. These characters are represented and stored as hexadecimal values 0D0A (CRLF) and 1A (EOF). Program source code defined with variable length records is the type of file generally sent with this option. Output records do not include the carriage return and line-feed characters.
L	Indicates that each record is preceded by a 2-byte hexadecimal record length. Select this option when sending a data set defined in a fixed format, or when sending binary data. The output record is determined by the value in the first 2 bytes that contain the record length.
N	Stores data as it is received. Output records are built based on the record length of the data set allocated to receive the data.

In this field:

Enter:

- Expedite/CICS, valid values are:

Value	Description
A	Stores each record with a carriage return and line-feed character and uses the end-of-file character. These characters are represented and stored as hexadecimal values 0D0A (CRLF) and 1A (EOF). Program source code defined with variable length records is the type of file generally sent with this option. Output records do not include the carriage return and line-feed characters.
L	Indicates that each record is preceded by a 2-byte hexadecimal record length. Select this option when sending a data set defined in a fixed format, or when sending binary data. The output record is determined by the value in the first 2 bytes that contain the record length.
O	Other. Free format.

- IBM Expedite/MVS Host Version 1.3 (TPMAIN), valid values are:

Value	Description
A	Stores each record with a carriage return and line-feed character and uses the end-of-file character. These characters are represented and stored as hexadecimal values 0D0A (CRLF) and 1A (EOF). Program source code defined with variable length records is the type of file generally sent with this option. Output records do not include the carriage return and line-feed characters.
B	Indicates that each record is preceded by a 2-byte hexadecimal record length. Select this option when sending a data set defined in a fixed format, or when sending binary data. The output record is determined by the value in the first 2 bytes that contain the record length.
C	Stores data as it is received. Output records are built based on the record length of the data set allocated to receive the data.

This field is not used when sending data or receiving EDI data. This field is only used when receiving free-form messages and files. If you specify a trading partner when requesting network activity (send or receive), the value for this field is taken from the trading partner profile. Otherwise, the value for this field is taken from the requestor profile.

IINR3 reference: STGFORM.

IINB1, IINB41, and IINB42 reference: DELIMITED.

Network cmds file

The name of a member of a PDS that will be allocated to the ddname of EDINTCMD. This member will contain the commands you want to pass to the networks supported in the future. DataInterchange will read the commands from the PDS member and write the commands to the *Network input file* specified in the network profile member after all substitutable variable tags have been resolved by DataInterchange.

Communications with Trading Partners

In this field:

Data line timeout

Enter:

A value the network can use as a maximum allowable time that the data line for communications can be idle without being dropped. If you specify a trading partner when requesting network activity (send or receive), the value for this field is taken from the TP profile. Otherwise, the value for this field is taken from the requestor profile.

Remote status pgm

This feature may be enabled in a future release of DataInterchange. The name of a program to be used to process network acknowledgments from a secondary network.

Note: When a gateway is used to connect to another VAN (such as IN), the gateway is referred to as the primary network because it is the network with which DataInterchange interfaces. The other VAN, such as IN, is referred to as the secondary or remote network because DataInterchange goes through the gateway network to reach the other network. When the gateway is used to connect directly to a trading partner's site or when the gateway is used as the only network, there is no secondary network.

This program will be used only if you are using a gateway to connect to a secondary network (assuming we call the gateway your primary network) and have requested and received network acknowledgments into (*Net acks file*) from the secondary network. If this is the case, specify the program name INB1MSG in this field.

Alt net dial num

The alternate phone number to dial to connect to your network.

Compression

A code that indicates whether or not Expedite Base/MVS should call a third party software program to compress the data before sending it to the network. If a TPNICKN keyword is specified on the PERFORM SEND command, the compression value comes from the trading partner profile. Otherwise, the compression value comes from the requestor profile.

Value Description

Y	Allows the data to be compressed before sending to the network
N	Leaves the data as is and does not compress (the default)
T	Uses the Expedite Base/MVS CPLOOKUP file to determine whether or not to compress the data

IINB42 reference: COMPRESS.

Priority

A blank (for normal priority) or a P (for high priority). These codes are used by Expedite Base/MVS and Expedite/CICS to prioritize delivery of messages. High priority messages will be delivered to your trading partners before normal priority messages.

For additional information, see the SEND commands in the *Expedite Base/MVS Programming Guide* or Chapter 6 in the *Customizing and Developing Applications with Expedite/CICS* manual.

Note: The priority value will come from the Trading Partner Profile, if a Trading Partner Profile member is used on the SEND. Otherwise, the priority value will come from the Requestor Profile.

5. Press Enter. The Add Profile Member panel (PM08) is redisplayed. You can add as many members as necessary for your organization. Press F3 (Exit) when you are done adding members.

Setting Up the Trading Partner Profile

The trading partner profile contains members that describe your trading partners. There is one member for each trading partner.

Adding a Trading Partner Profile Member

To add a member to the trading partner profile, follow these steps:

1. From the Administrator's Menu, select *Profiles*. The Profile Definitions panel (PM01) is displayed.
2. Before adding a member, you can view the profile. Type **V** in the action column next to TPPROF, and press Enter.

Log work with Members Print View			
PM01		Profile Definitions	1 to 17 of 17
Action	Profile ID	Profile Description	Log?
—	ACTLOGS	Activity log	N
—	ADAMCTL	User program information	N
—	APPDEFS	Application Definition Profile	N
—	CONTRECV	Continuous Receive Profile	N
—	E	EDIFACT standard envelope data	N
—	I	ICS standard envelope data	N
—	LANGPROF	Language profile	N
—	MQSERIES	MQSeries Queue Profile	N
—	NETOP	Network operation profile	N
—	NETPROF	Network profile	N
—	REQPROF	Requestor profile	N
—	SECUPROF	Security profile	N
—	SYSPROF	System profile	N
—	T	UN/TDI standard envelope data	N
V	TPPROF	Trading partner profile	Y
—	U	UCS standard envelope data	N
—	X	X12 standard envelope data	N

The View Profile Definition panel (PM05) is displayed.

PM05		View Profile Definition		1 to 14 of 64
Profile ID: TPPROF		Profile description: Trading partner profile		
Key==>	Field Label	Length	Type	Description
	TP nickname	016	CH	Trading partner nickname
	Network ID	008	CH	Network ID
	System qualifier	001	CH	Intersystem address qualifier
	System ID	008	CH	Intersystem ID
	Account number	032	CH	Network account number
	User ID	032	CH	Network user ID
	Interchange qualif	004	CH	Interchange qualifier
	Interchange ID	035	CH	Interchange sender/receiver ID
	Company name	040	CH	Company name
	Address line 1	040	CH	Company address line 1
	Address line 2	040	CH	Company address line 2
	Address line 3	040	CH	Company address line 3
	City	030	CH	City
	State	002	CH	State

Communications with Trading Partners

PM05 View Profile Definition 14 to 27 of 64

Profile ID: TPPROF Profile description: Trading partner profile

Field Label	Length	Type	Description
State	002	CH	State
Postal code	015	CH	Postal code
Country	030	CH	Country
Contact phone	025	CH	Contact phone number
Contact fax	025	CH	Contact fax number
Contact name	030	CH	Contact name
Interchng send PW	014	CH	Interchange password for send
Interchng recv PW	014	CH	Interchange password for receive
Security ID	008	CH	Name of security profile member
Network msg class	001	CH	Network classification
Network charges	001	CH	Network charge code
Net acknowledgment	001	CH	Network acknowledgment code
Destination verif	001	CH	Destination verification code
Retention period	003	CH	Mailbox retention period

PM05 View Profile Definition 27 of 40 of 64

Profile ID: TPPROF Profile description: Trading partner profile

Field Label	Length	Type	Description
Retention period	003	CH	Mailbox retention period
EDI option	001	CH	Option for storing received data
EDI proc override	001	CH	Special processing, received data
Format override	001	CH	Storage format override
Machine type	001	CH	Machine type
Storage format	001	CH	Storage format
End of text/msg	001	CH	End of text/message delimiter
Log standard data	001	CH	Log standard data (Y/N)
Functional group	001	CH	Send functional group (Y/N)
Subelement delim	001	HX	Sub-element delimiter
Data element delim	001	HX	Data element delimiter
Segment delimiter	001	HX	Segment delimiter
Seg ID separator	001	HX	Segment ID separator
Decimal notation	001	CH	Decimal notation

PM05 View Profile Definition 40 to 53 of 64

Profile ID: TPPROF Profile description: Trading partner profile

Field Label	Length	Type	Description
Decimal notation	001	CH	Decimal notation
Release character	001	HX	Release character
Interchange mask	009	CH	Interchange control number mask
Group mask	009	CH	Group control number mask
Transaction mask	009	CH	Transaction control number mask
Comment line 1	040	CH	Comment line 1
Comment line 2	040	CH	Comment line 2
Comment line 3	040	CH	Comment line 3
Comment line 4	040	CH	Comment line 4
Comment line 5	040	CH	Comment line 5
Comment line 6	040	CH	Comment line 6
Comment line 7	040	CH	Comment line 7
Comment line 8	040	CH	Comment line 8
Comment line 9	040	CH	Comment line 9

PM05	View Profile Definition			53 to 64 of 64
Profile ID: TPPROF		Profile description: Trading partner profile		
	Field Label	Length	Type	Description
	Comment line 9	040	CH	Comment line 9
	Comment line 10	040	CH	Comment line 10
	Network cmds file	008	CH	Network commands file
	TP data phone num	032	CH	Trading Partner data phone line
	Data line timeout	004	CH	Data line communication timeout
	Segmented output	001	CH	Segmented output indicator
	File suffix	002	CH	File suffix for fixed translations
	Env profile suffix	002	CH	Envelope prof member name suffix
	Allow generic recv	001	CH	Allow generic receive usage (Y/N)
	Compression	001	CH	Compression tag (T/Y/N)
	Priority	001	CH	Delivery priority (blank/P)
	Partner type	001	CH	Trading partner type (E/A/B)

The *Field Labels* are the prompts you see when providing data for a member of this profile. The first label is the *key* or name of this field. The next two columns show the characteristics of the data you can use. The last column contains a brief description of the fields.

- Press F3 (Exit) when you are finished viewing the profile definition. The Profile Definitions panel (PM01) is redisplayed.
- Type **M** in the action column next to TPPROF, and press Enter. The Profile Members panel (PM07) is displayed.
- Type **A** in the action column next to any item, and press Enter.

Add Copy Delete List Print Update coNtrol numbers View				
PM01	Profile Definitions			1 to 17 of 17
Action	Profile ID	PM07 Profile Members 1 to 9 of 54		
-	ACTLOGS	Profile ID : TPPROF Description: Trading partner profile Action Key a AAASPORTSCLOTHES - AABSPORTSCLOTHES - AACSCCTSPORTSCLOTHES - AADSPORTSCLOTHES - AAESPORTSCLOTHES - AL'S - AVNT - BBDECTP - BBDEMOTP		
-	ADAMCTL			
-	APPDEFS			
-	CONTRECV			
-	E			
-	I			
-	LANGPROF			
-	MQSERIES			
-	NETOP			
-	NETPROF			
-	REQPROF			
-	SECUPROF			
-	SYSPROF			
-	T			
m	TPPROF			
-	U			
-	X	X12 standard envelope data N		

The Add Profile Member panel (PM08) is displayed.

Communications with Trading Partners

PM08 Add Profile Member 1 to 13 of 64

Profile ID: TPPROF

Fill in the information below and press Enter to save this member.
To stop entering members, press Exit or Cancel.

TP nickname
Network ID
System qualifier
System ID
Account number
User ID
Interchange qualif
Interchange ID
Company name
Address line 1
Address line 2
Address line 3
City

PM08 Add Profile Member 13 to 25 of 64

Profile ID: TPPROF

Fill in the information below and press Enter to save this member.
To stop entering members, press Exit or Cancel.

City
State
Postal code
Country
Contact phone
Contact fax
Contact name
Interchng send PW
Interchng rcv PW
Security ID
Network msg class
Network charges
Net acknowledgment

PM08 Add Profile Member 25 to 37 of 64

Profile ID: TPPROF

Fill in the information below and press Enter to save this member.
To stop entering members, press Exit or Cancel.

Net acknowledgment
Destination verif
Retention period
EDI option
EDI proc override
Format override
Machine type
Storage format
End of text/msg
Log standard data
Functional group
Subelement delim 00
Data element delim 00

PM08	Add Profile Member	37 to 49 of 64
Profile ID: TPPROF		
Fill in the information below and press Enter to save this member. To stop entering members, press Exit or Cancel.		
Data element delim	00	
Segment delimiter	00	
Seg ID separator	00	
Decimal notation		
Release character	00	
Interchange mask	_____	
Group mask	_____
Transaction mask	.	_____
Comment line 1	. .	_____
Comment line 2	. .	_____
Comment line 3	. .	_____
Comment line 4	. .	_____
Comment line 5	. .	_____

PM08	Add Profile Member	49 to 61 of 64
Profile ID: TPPROF		
Fill in the information below and press Enter to save this member. To stop entering members, press Exit or Cancel.		
Comment line 5	. .	_____
Comment line 6	. .	_____
Comment line 7	. .	_____
Comment line 8	. .	_____
Comment line 9	. .	_____
Comment line 10	. .	_____
Network cmds file		_____
TP data phone num		_____
Data line timeout		_____
Segmented output		_____
File suffix	_____
Env profile suffix		_____
Allow generic rcv		_____

PM08	Add Profile Member	61 to 64 of 64
Profile ID: TPPROF		
Fill in the information below and press Enter to save this member. To stop entering members, press Exit or Cancel.		
Allow generic rcv		_____
Compression	_____
Priority	_____
Partner type	_____

6. Type the appropriate information in each field, and press Enter. Press F8 (Forward) to scroll to the remaining panels. If you need additional information about the use of a specific field, place the cursor in that field and press F1 (Help).

Note: Trading partner profile values override requestor profile values. Trading partner transactions (maps) can reside physically in the DataInterchange host database or in the DataInterchange Client database. Mappings which reside in the DataInterchange Client database are identified in the TPT list by special flags. These mappings cannot be updated using the DataInterchange host interface. For more information about using the DataInterchange client interface, See “Using DataInterchange with the DataInterchange Client Graphical User Interface” on page 1-13.

Communications with Trading Partners

The following list describes the information that you type in each of the profile member fields.

In this field:

TP nickname

Enter:

The name you use to refer to this trading partner. Use the same name throughout DataInterchange to refer to this trading partner by "nickname." This field is required.

Network ID

The name of the network used to communicate with this trading partner. This field is required. This ID must match the name of a network profile member, such as IINB41. If you are translating and sending data from both MVS and CICS, see "Handling a Trading Partner in Multiple Environments" on page 5-42 for more details about the value of this field.

System qualifier

I to indicate that intersystem addressing is required. This field applies only to IN.

IINR3 reference: DTBLTYP.

System ID

The ID of the system responsible for the receiver's account. Valid values are:

ID	Description
----	-------------

EUR	IBM Global Network Europe serves customers in Europe, the United Kingdom, and Hong Kong.
-----	--

JPN	IBM Global Network Japan serves customers in Japan.
-----	---

USA	IBM Global Network serves customers in North America, including Canada.
-----	---

USQ	IBM Global Network Quick Start System currently serves customers in Australia, New Zealand, and Latin America. These countries are subject to change as new countries are moved to their own Information Exchange System.
-----	---

IINR3 reference: DTBLID.

IINB1, IINB41, and IINB42 reference: SYSID.

Account number

Your trading partner's network account number or account ID. If entered, the combination of *Account number* and *User ID* must be unique for each member. The account number must be left-justified. For sending and receiving standard transactions using the IN, the account number can be no longer than seven characters.

For transactions that you are sending to this trading partner, if *Interchange ID* is blank, the *Account number* and *User ID* make up the receiver ID in the interchange envelope (which defines the mailbox where transactions for this trading partner are delivered). Type U (BG/EG) envelopes are an exception (see *Contact phone* field).

If the *Interchange ID* is blank, the *Account number* may be needed to process network acknowledgments. See the *DataInterchange Programmer's Reference* for more information.

In this field:

User ID

Enter:

Your trading partner's network user ID. If entered, the combination of *Account number* and *User ID* must be unique for each member. The account number must be left-justified. For sending and receiving standard transactions using the IN, the account number can be no longer than seven characters.

For transactions that you are sending to this trading partner, if *Interchange ID* is blank, the *Account number* and *User ID* make up the receiver ID in the interchange envelope (which defines the mailbox where transactions for this trading partner are delivered). Type U (BG/EG) envelopes are an exception (see *Contact phone* field).

If the *Interchange ID* is blank, the *User ID* may be needed to process network acknowledgments. See the *DataInterchange Programmer's Reference* for more information.

Interchange qualif

The type of interchange ID in the *Interchange ID* field. These codes are defined in the EDI standard. A value of 01 indicates a Dun & Bradstreet (DUNS) number. If the *Interchange ID* is blank, the enveloper takes the qualifier from the envelope profile member. The combination of this field and the *Interchange ID* field must be unique for each member unless they are blank.

Provide a value in the *Interchange qualif* and *Interchange ID* fields of each trading partner profile member for the best response during any send or receive process.

Interchange ID

The interchange receiver ID when sending to this partner, and the interchange sender ID when receiving from this partner. For example, this could be a DUNS number, an account number and user ID, or a telephone number.

The *Interchange qualif* and *Interchange ID* fields provide the data used to build or match the interchange receiver ID field in the envelope segment. These fields are optional, and if not provided, the enveloper uses the account number and user ID for type E (UNB/UNZ), I (ICE/ICS), T (STX/END), and X (ISA/IEA) envelopes, or the contact phone for type U (BG/EG) envelopes. If provided, the combination of interchange qualifier and interchange ID must be unique.

If *Account number* is blank, the *Interchange ID* may be needed to process network acknowledgments. See *DataInterchange Programmer's Reference* for more information.

Provide a value in the *Interchange qualif* and *Interchange ID* fields of each trading partner profile member for the best response during any send or receive process.

Company name

The name of the trading partner's company. The company name may be used as envelope data by using the 'CO' envelope data type.

Address line 1

Line 1 of the trading partner's address.

Address line 2

Line 2 of the trading partner's address.

Address line 3

Line 3 of the trading partner's address.

City

City of the trading partner's city.

State

State of the trading partner's state.

Communications with Trading Partners

In this field:	Enter:
Postal Code	Postal or zip code of the trading partner's city and state.
Country	Country of the trading partner.
Contact phone	The trading partner's telephone number. For type U (BG/EG) enveloping, the contact phone number is used as the interchange receiver ID if the <i>Interchange ID</i> field is blank. When deenveloping type U (BG/EG) interchanges, the contact phone number is used as the interchange sender ID if the <i>Interchange ID</i> field is blank.
Contact fax	Fax number of the trading partner.
Contact name	The name of the person you speak with when dealing with this trading partner.
Interchnng send PW	The value used as a password in the interchange envelope when sending data to this trading partner. Before sending any data, you and your trading partner should agree on a value for this field. This value will be used in the interchange envelope data element that has a data type of PW.
Interchnng rcv PW	The value expected as a password in the interchange envelope when receiving data from this trading partner. If this value matches the password that is in received interchange envelope, then translation occurs. Otherwise, the translator logs an error.
Security ID	The name of the default security profile member (SECUPROF) that specifies the encryption and authentication processes used for this trading partner. This profile member is always used when receiving from this partner and is used when sending to this partner, unless the send usage for the transaction specifies a different member in either the <i>Group security profile member name</i> field or the <i>Trans security profile member name</i> field.
Network msg class	<p>A code that indicates any special status of the data being sent. For IBM Global Network, T indicates test status and a blank indicates normal status.</p> <p>This field is not used when receiving data. If you specify a trading partner when requesting network activity (send or receive), the value for this field is taken from the trading partner profile. Otherwise, the value for this field is taken from the requestor profile.</p> <p>IBM Global Network reference: MSGNCLS.</p>

In this field:
Network charges

Enter:

The charge code for network charges. This code allows the network to determine who should be charged when data is sent. Valid values are:

Code	Description
1	Receiver pays all charges.
2	Receiver pays all charges by agreement. If receiver has not agreed to pay all charges, charges are split between the sender and receiver.
3	Receiver pays all charges by agreement. If receiver has not agreed to pay all charges, charges are split between the sender and receiver, by agreement, or the sender pays all charges. Code 3 is the default for the Network charges field.
4	Charges are split between the sender and receiver by agreement, or the sender pays all charges.
5	Charges are split between the sender and receiver.
6	Sender pays all charges.

This field is not used when receiving data. If you specify a trading partner when requesting network activity (send or receive), the value for this field is taken from the trading partner profile. Otherwise, the value for this field is taken from the requestor profile.

IINR3 reference: MSGCHRG.

IINB1, IINB41, and IINB42 reference: CHARGE.

For a detailed explanation of IBM Global Network message charges, see the *Understanding Information Exchange Message Charges*.

Net acknowledgment

The code that indicates which network acknowledgments, such as received by network (receipt), delivered to trading partner (delivery), or purge, that you want returned to you when sending data to trading partners. Valid values are:

Code	Description
(blank)	No acknowledgments
R	Receipt acknowledgments only
D	Delivery acknowledgments only
B	Both receipt and delivery acknowledgments
A	Purge acknowledgments only
C	Both receipt and purge acknowledgments
E	Either purge or delivery acknowledgments
F	Receipt acknowledgments, and either delivery or purge acknowledgments

This field is not used when receiving data. If you specify a trading partner when requesting network activity (send or receive), the value for this field is taken from the trading partner profile. Otherwise, the value for this field is taken from the requestor profile.

IINR3 reference: MSGRCPT.

IINB1, IINB41, and IINB42 reference: ACK.

Communications with Trading Partners

In this field:

Destination verif

Enter:

The code that indicates whether the destination and authorization are to be verified before sending. Valid values are:

Code	Description
------	-------------

N	No verification (default)
---	---------------------------

Y	Verification
---	--------------

F	Verification and sending even if the destination is not verified (useful for intersystem addressing)
---	--

This field is not used when receiving data. If you specify a trading partner when requesting network activity (send or receive), the value for this field is taken from the trading partner profile. Otherwise, the value for this field is taken from the requestor profile.

IINR3 reference: MSGVCHK.

IINB1, IINB41, and IINB42 reference: VERIFY.

Retention period

The number of days that data is kept in a mailbox before it is purged. For IBM Global Network, enter blanks or 000 to use the default, or enter a range from 1 to 180.

This field is not used when receiving data. If you specify a trading partner when requesting network activity (send or receive), the value for this field is taken from the trading partner profile. Otherwise, the value for this field is taken from the requestor profile.

IINR3 reference: MSGRETN.

IINB1, IINB41, and IINB42 reference: RETAIN.

EDI option

Y if you want to store EDI segments as separate records, or **N** if you do not. The DataInterchange translator accepts data in either format.

This field is used when receiving or sending data. If you specify a trading partner when requesting network activity, the value for this field is taken from the trading partner profile. Otherwise, the value for this field is taken from the requestor profile.

IBM Global Network reference: EDIOPT.

EDI proc override

A code that indicates whether or not data that you receive is to have special EDI processing, which consists of breaking the records by the segment delimiter. For IBM Global Network, valid values are:

Code	Description
------	-------------

Y	Perform EDI processing if the common data header indicates that the data in the file is in EDI standard format.
---	---

N	Omit EDI processing, regardless of the common data header.
---	--

This field is not used when sending data. If you specify a trading partner when requesting network activity (send or receive), the value for this field is taken from the trading partner profile. Otherwise, the value for this field is taken from the requestor profile.

IINR3 reference: EDIPROC.

IINB1, IINB41, and IINB42 reference: AUTOEDI.

In this field:

Format override

Enter:

Y if you want to use the storage format defined in the common data header, or **N** if you do not.

The common data header provides details (such as file name, carriage-return, and line-feed options) that let the receiving interface reconstruct a received message into its original format. It also makes more information available to the recipient of a file. If there is no common data header, the format indicated in the *Storage format* field is used.

This field is not used when sending data. If you specify a trading partner when requesting network activity (send or receive), the value for this field is taken from the trading partner profile. Otherwise, the value for this field is taken from the requestor profile.

IINR3 reference: STGFORMO.

IINB1, IINB41, and IINB42 reference: DLMOVERRIDE.

Machine type

1 if your trading partner is using the Personal Computer/Information Exchange (PC/IE) product. Otherwise, leave the field blank.

Storage format

The code that tells the network how data is stored for free-form messages and files.

Consider the type of data you want to send and how the file is received when determining what values to select. Contact a representative from each network you are using for all available values.

For example, if you are using:

- Expedite/Base MVS Version 1.1 (IEBASE), valid values are:

Value	Description
-------	-------------

- | | |
|---|---|
| C | Stores each record with a carriage return and line-feed character and uses the end-of-file character. These characters are represented and stored as hexadecimal values 0D0A (CRLF) and 1A (EOF). Program source code defined with variable length records is the type of file generally sent with this option. Output records do not include the carriage return and line-feed characters. |
| L | Indicates that each record is preceded by a 2-byte hexadecimal record length. Select this option when sending a data set defined in a fixed format, or when sending binary data. The output record is determined by the value in the first 2 bytes that contain the record length. |
| N | Stores data as it is received. Output records are built based on the record length of the data set allocated to receive the data. |

In this field:

Enter:

- Expedite/CICS, valid values are:

Value	Description
A	Stores each record with a carriage return and line-feed character and uses the end-of-file character. These characters are represented and stored as hexadecimal values 0D0A (CRLF) and 1A (EOF). Program source code defined with variable length records is the type of file generally sent with this option. Output records do not include the carriage return and line-feed characters.
L	Indicates that each record is preceded by a 2-byte hexadecimal record length. Select this option when sending a data set defined in a fixed format, or when sending binary data. The output record is determined by the value in the first 2 bytes that contain the record length.
O	Other. Free format.

- IBM Expedite/MVS Host Version 1.3 (TPMAIN), valid values are:

Value	Description
A	Stores each record with a carriage return and line-feed character and uses the end-of-file character. These characters are represented and stored as hexadecimal values 0D0A (CRLF) and 1A (EOF). Program source code defined with variable length records is the type of file generally sent with this option. Output records do not include the carriage return and line-feed characters.
B	Indicates that each record is preceded by a 2-byte hexadecimal record length. Select this option when sending a data set defined in a fixed format, or when sending binary data. The output record is determined by the value in the first 2 bytes that contain the record length.
C	Stores data as it is received. Output records are built based on the record length of the data set allocated to receive the data.

This field is not used when sending data or receiving EDI data. This field is only used when receiving free-form messages and files. If you specify a trading partner when requesting network activity (send or receive), the value for this field is taken from the trading partner profile. Otherwise, the value for this field is taken from the requestor profile.

IINR3 reference: STGFORM.

IINB1, IINB41, and IINB42 reference: DELIMITED.

End of text/msg

The character that signifies to the network the end-of-data for free-form messages and data files. This value does not apply to standard transactions.

IINR3 reference: EORCHAR.

IINB1, IINB41, and IINB42 reference: ENDSTR.

In this field:	Enter:
Log standard data	Y if you want DataInterchange to write an image of any envelope created for sending to or received from this trading partner, or N if you do not. The setting of this field controls the logging of standard data only when the Log standard data field of the APPDEFS profile member is not a Y or an N.
Functional group	Y if you want to create functional groups for transactions with type E (UNB/UNZ) envelopes, N if you do not. Functional groups are always created for type I (ICS/ICE), U (BG/EG), and X (ISA/IEA) envelopes, and they are never created for type T (STX/END) envelopes.
Complete the next six fields only if you want to override the values in the envelope standard.	
<ul style="list-style-type: none"> Each delimiter or character must be different from the other five. However, the subelement delimiter and the data element delimiter can be the same for I (ICS/ICE), U (BG/EG), and X (ISA/IEA) type envelopes. These fields are used only when interchanges are created for sending. When interchanges are received, the delimiters are extracted from segments and fields within the received interchange. If you use any of these fields with type E (UNB/UNZ) envelopes, a UNA segment is generated when the interchange is created. If you use any of these fields with type T (STX/END) envelopes, a SCH segment is generated when the interchange is created. 	
Subelement delim	The hexadecimal value of the character that separates subelements (component data elements) in a transaction set. An entry here (other than 00 or 40) overrides the character specified in the standard.
Data element delim	The hexadecimal value of the character that separates the data elements of a transaction set. An entry here (other than 00 or 40) overrides the character specified in the standard. The value is only used when interchanges are created. For interchanges received, the delimiters are extracted from segments, or fields, or both within the interchange.
Segment delimiter	The hexadecimal value of the character that marks the end of each segment in a transaction set. An entry here (other than 00 or 40) overrides the character specified in the standard. The value is only used when interchanges are created. For interchanges received, the delimiters are extracted from segments, or fields, or both within the interchange.
Seg ID separator	The hexadecimal value of the character that separates the segment ID and the first data element in a segment for type E (UNB/UNZ) envelopes only. An entry here (other than 00 or 40) overrides the character specified in the standard.
Decimal notation	The character that represents decimal points in a transaction set. For type E (UNB/UNZ) envelopes, a value in this field overrides the character specified in the standard. For all other types, a period represents the decimal point. The value is only used when interchanges are created. For interchanges received, the delimiters are extracted from segments, or fields, or both within the interchange.

Communications with Trading Partners

In this field:

Release character

Enter:

For type E (UNB/UNZ) and T (STX/END) envelopes, specifies the hexadecimal value of the character that indicates when a delimiter is being used as part of the data. An entry here (other than 00 or 40) overrides the character specified in the standard. The value is only used when interchanges are created. For interchanges received, the delimiters are extracted from segments, or fields, or both within the interchange.

Interchange mask

The initial reference number that the enveloper maps to the CN data type in the interchange header and trailer. This value will be used as the base value for each trading partner, receiver ID combination. This field does not represent the current control number for this trading partner. Use the control number action from the member list panel to request control number information.

Group mask

The initial reference number or special codes that the enveloper maps to the CN data type in the functional group header and trailer. This value will be used as the base value for each trading partner, receiver ID combination. Use caution when updating this field. This value does not represent the current control number for this trading partner. Use the control number action from the member list panel to request control number information.

Transaction mask

The initial reference number or special codes that the enveloper maps to the CN data type in the transaction set header and trailer. This value will be used as the base value for each trading partner, receiver ID combination. Use caution when updating this field. This value does not represent the current control number for this trading partner. Use the control number action from the member list panel to request control number information.

Comment line 1
through 10

Free-form notes about the trading partner.

Network cmds file

The name of a member of a PDS that will be allocated to the ddname of EDINTCMD. This member will contain the commands that you want to pass to the Gateway. DataInterchange will read the commands from the PDS member and write the commands to the *Network input file* specified in the network profile member after all substitutable variable tags have been resolved by DataInterchange.

TP data phone num

The phone number that the network program can use to dial to connect to your trading partner directly. This is the number needed by your computer to talk directly to your trading partner's computer. See the *Contact phone* field for the voice number for you (not your computer) to call a person at your trading partner's site.

Data line timeout

A value that the network can use as a maximum allowable time that the data line for communications can be idle without being dropped. If you specify a trading partner when requesting network activity (send or receive), the value for this field is taken from the TP profile. Otherwise, the value for this field is taken from the requestor profile.

Segmented output

A code that indicates whether or not you want EDI segments to be stored in the output file as separate records.

Value

Description

Y

Ends records at the segment delimiter

N

Does not end records at the segment delimiter (the default)

In this field:

File suffix

Enter:

A 2-character suffix that will be used as a suffix for the ddname used to store the results of a Fixed-to-Fixed translation. The basic part of the ddname is taken from the *Application file name* field of the target application data format.

This 2-character suffix will be appended to the *Application file name* or will overlay the last 2 characters of the *Application file name* field if that value exceeds 6 characters. The suffix is provided so that the results of a Fixed-to-Fixed translation can be separated by the trading partner. If this separation is not wanted, then either the suffix can be left blank or all the ddnames can be assigned to the same physical file.

In a CICS environment, the suffix can be used to identify a unique TS queue for each trading partner.

Env profile suffix

A 2-character suffix that will be used as a suffix for a generic standard envelope profile member name.

Allow generic rcv

A code that indicates whether or not a generic receive usage can be used to translate transactions from this trading partner.

Value Description

Y Allows translation by generic receive usages for this trading partner

Other Does not allow translation by generic receive usages for this trading partner

See “Defining Generic Receive Usages” on page 9-96 for more information.

Compression

A code that indicates whether or not Expedite Base/MVS should call a third party software program to compress the data before sending it to the network. If a TPNICKN keyword is specified on the PERFORM SEND command, the compression value comes from the trading partner profile. Otherwise, the compression value comes from the requestor profile.

Value Description

Y Allows the data to be compressed before sending to the network

N Leaves the data as is and does not compress (the default)

T Uses the Expedite Base/MVS CPLOOKUP file to determine whether or not to compress the data

IINB42 reference: COMPRESS.

Priority

A blank (for normal priority) or a P (for high priority). These codes are used by Expedite Base/MVS and Expedite/CICS to prioritize delivery of messages. High priority messages will be delivered to your trading partners before normal priority messages.

For additional information see the SEND commands in the *Expedite Base/MVS Programming Guide* or Chapter 6 in the *Customizing and Developing Applications with Expedite/CICS* manual.

Note: The priority value will come from the Trading Partner Profile, if a Trading Partner Profile member is used on the SEND. Otherwise, the priority value will come from the Requestor Profile.

Communications with Trading Partners

In this field:

Partner type

Enter:

Enter a value that indicates the type of trading partner this definition represents.

Valid values are:

Value	Description
-------	-------------

E	EDI trading partner
---	---------------------

An EDI trading partner is a trading partner considered to be external to your business organization. The interchange control numbers are generated using the EDI trading partner. This type of trading partner is the default and will have no effect on your current trading partner setup.

A	Application trading partner
---	-----------------------------

An application trading partner is considered an internal trading partner within your business organization. For example, divisions or departments may be considered internal trading partners. The application trading partner may be used when no specific EDI trading partner is defined or in combination with the EDI trading partner. The interchange control numbers are generated using the application and EDI trading partner combination. This type of trading partner should be used with centralized EDI, when the application trading partners do business with the same EDI trading partner.

B	Both EDI trading partner and application trading partner
---	--

This trading partner is considered both an external (EDI) and internal (application) trading partner. This type of trading partner should be used when your organization provides EDI translation services to customers who are trading partners.

The value of this field is directly related to the trading partner usage setup.

More information may be found in the *DataInterchange Programmer's Reference*. See also "Migrating Trading Partner Usages" on page 9-101.

7. Press Enter. The Add Profile Member panel (PM08) is redisplayed. You can add as many members as necessary for your organization. Press F3 (Exit) when you are done adding members. The Profile Definitions panel (PM01) is displayed.
8. Normally the Translator will create trading partner control number records for your trading partner. Optionally, to predefine the pairings for this trading partner, complete steps 9 through 12 below.
9. Type **M** in the action column next to TPPROF, and press Enter. The Profile Members panel (PM07) is redisplayed. Type **N** in the action column next to the trading partner for which you want to define a control number pairing. The Add Trading Partners Control Numbers panel (PM15) is displayed.
10. Enter the Receiver ID and Receiver Qualifier. Optionally, enter the Standard Transaction ID. Then, either set the Interchange Group and Transaction Control Number fields to zero or to an appropriate control number mask (see Table 5-2 on page 5-40).

Press Enter. The Trading Partner Control Numbers panel (PM14) is displayed. The new pairing should be displayed on this panel.
11. To create additional control number pairings, type **A** in the "A" (action) column. The Add Trading Partner Control Numbers panel (PM15) is redisplayed. Repeat step 10.

12. Once you have finished entering control number pairings, press F3 to return to the Profile Definitions panel (PM01).

Note: When you have defined your network by setting up the network profile, defined yourself by setting up the requestor profile, and defined your trading partners by adding members to the trading partner profile, you are ready to begin communicating with your trading partners. The rest of this chapter describes features needed for complex communication situations. If your communication needs do not require these features, proceed to the next chapter.

Utilizing Control Numbers

DataInterchange provides customers the ability to assign interchange, group, and transaction level control numbers by sender ID, sender qualifier, receiver ID, receiver qualifier, and standard transaction ID. This creates a one-to-many relationship between senders and receivers of EDI transactions. The sender has the ability to customize the way the interchange, group, and transaction control numbers are generated per receiver.

A Trading Partner Control Number list panel (PM14) and Add Trading Partner Control Numbers entry panel (PM15) allow the maintenance of control numbers (see below). The panels are accessible through the Profile Members panel (PM07). The Add Trading Partner Control Numbers panel is invoked from the Trading Partner Control Numbers list panel by typing an **N** in the action column of the desired Trading Partner Profile member on PM07. If any control number pairing exists, the list panel will be shown. If no control number pairings exist, the Add Trading Partner Control Numbers panel (PM15) will be shown. The control numbers (or control number masks, if used in lieu of last used control numbers) associated with the pairing can be updated by selecting the desired pairing and typing a **U** in the A (action) column. The Update Trading Partner Control Numbers panel (PM17) is displayed, with the fields that can be changed left unprotected.

Add Copy Delete List Print Update View			
PM01		Profile Definitions 1 to 15 of 17	
Action	Profile ID	PM07 Profile Members 1 to 9 of 245	
-	ACTLOGS	Profile ID : TPPROF	
-	ADAMCTL	Description: Trading partner profile	
-	APPDEFS		
-		PM14 Trading Partner Control Numbers 1 to 2 of 5	
-		TP nickname : DIBBLE-COOKIES	
-		TP type : E (E = EDI, A = Application, B = Both)	
-			
-	A Receiver ID	Receiver Qualifier	Standard Trans ID
-	DIBCOOK	0014	VER0257X
-	DIBCOOKAUSTIN	0002	X0535112
m	TPPROF		

PM15 Add Trading Partner Control Numbers	
TP nickname	DIBBLE-COOKIES
Receiver ID	DIBCOOKAUSTIN
Receiver qualifier	0002
Standard transaction ID	X0535112
Interchange control number	000000001
Group control number	000000034
Transaction control number :	000000022

If no applicable control number record is found during translation, the translator will create a control number record for the sender/receiver pairing, assign control numbers for the first translated message, and increment the control numbers for subsequent translations.

Assigning Control Numbers by Standard Transaction ID

Control numbers can be assigned by sender/receiver pairing or by standard transaction ID within the sender/receiver pairing. The mapping usage must indicate that control numbers are to be assigned by standard transaction ID. (See Chapter 9, “Mapping Your Application Data to an EDI Standard Transaction Set,” for more information on setting the control number indicator.) If this is so indicated, the translator will create a control number record for the sender/receiver pair, and the standard transaction ID for the first translated message. It will use this record to increment the control numbers for subsequent translations of the same usage.

Group Control Number Masks

To modify the normal control number processing of DataInterchange, you can use special codes in the group control and transaction control fields of the trading partner profile member. Table 5-2 describes the special control codes for the group or transaction control numbers.

Table 5-2 (Page 1 of 2). Special Codes for Group Control Number Masks

Code	Control Number	Description
G	Transaction	The transaction control number is the same as the group control number. Only one transaction for each group is allowed.
Gn	Transaction	<i>n</i> bytes are taken from the group control number field. The remainder of the transaction control number is padded with zeros to its maximum size. Only one transaction for each group is allowed.
C	Group, Transaction	The remaining bytes in the group or transaction control number field are used to maintain a control number for this trading partner.
V	Group, Transaction	An incrementing value is used so that the first group or transaction has a value of 1; the second, a value of 2; and so on.
Vn	Transaction	An incrementing value <i>n</i> bytes long is used so that the first transaction has a value of 1; the second, a value of 2; and so on.
GnC	Transaction	<i>n</i> bytes are taken from the group control number and the remaining bytes in the transaction control number field are used to maintain a control number. The number of positions left determines the maximum value of the control number. For example, G5C leaves four positions; therefore, the maximum value is 9999. The control number cycles from the maximum number to 1.
GnV	Transaction	<i>n</i> bytes are taken from the group control number. For the remaining bytes in the transaction control number field, an incrementing value is used so that the first transaction has value of 1; the second, a value of 2; and so on.
GnVm	Transaction	<i>n</i> bytes are taken from the group control number. For the remaining bytes in the transaction control number field, an incrementing value is used so that the first transaction has a value of 1; the second, a value of 2; and so on.
I	Group, Transaction	The group or transaction control number should be the same as the interchange control number. Only one group is allowed for the interchange, and only one transaction is allowed for the group or interchange.
In	Group, Transaction	<i>n</i> bytes are taken from the interchange control number field. The remainder of the group or transaction control number is padded with zeros to its maximum size. Only one group is allowed for each interchange, and only one transaction is allowed for each group.

Table 5-2 (Page 2 of 2). Special Codes for Group Control Number Masks

Code	Control Number	Description
InC	Group, Transaction	<p>n bytes are taken from the interchange control number. The remaining bytes in the group or transaction control number field are used to maintain a control number.</p> <p>The number of positions left determines the maximum value of the control number. For example, I5C leaves four positions; therefore, the maximum value is 9999. The control number cycles from the maximum number to 1.</p>
InV	Group, Transaction	<p>n bytes are taken from the interchange control number. For the remaining bytes in the group or transaction control number field, an incrementing value is used so that the first group or transaction has a value of 1; the second, a value of 2; and so on.</p>
InVm	Transaction	<p>n bytes are taken from the interchange control number. For the remaining bytes, up to m bytes in the transaction control number field, an incrementing value is used so that the first transaction has a value of 1; the second, a value of 2; and so on.</p>
InGm	Transaction	<p>n bytes are taken from the interchange control number field, and a maximum of m bytes are taken from the group control number.</p> <p>If the n plus m is greater than 9, then only $9 - n$ bytes are taken from the group control number. For example, using I4G6, 4 bytes are taken from the interchange control number, and 5 bytes ($9 - 4$) are taken from the group control number. If the n plus m is less than 9, the remainder of the transaction control number is padded with zeros.</p> <p>Only one transaction for each group is allowed.</p>
InGmC	Transaction	<p>n bytes are taken from the interchange control number field and m bytes are taken from the group control number. The remaining bytes in the transaction control number field are used to maintain a control number.</p> <p>The number of positions left determines the maximum value of the control number. For example, I2G4C leaves three positions; therefore, the maximum value is 999. The control number cycles from the maximum number to 1.</p>
InGmV	Transaction	<p>n bytes are taken from the interchange control number field and m bytes are taken from the group control number. For the remaining bytes in the transaction control number field, an incrementing value is used so that the first transaction has value of 1; the second, a value of 2; and so on.</p>
InGmVo	Transaction	<p>n bytes are taken from the interchange control number field and m bytes are taken from the group control number. For the remaining bytes, up to o bytes in the transaction control number field, an incrementing value is used so that the first transaction has a value of 1; the second, a value of 2; and so on.</p>

Notes:

1. No error checking is done on these fields; therefore, no warning messages are generated for incorrect values. For example, if you enter I2G3C in the group control number field, the system processes the value as if you entered I2C. G3 is ignored because it is not valid for the group control number. Processing of the control field ends as soon as the maximum number of bytes for the control number has been reached. For example, if you enter I5G5V for the transaction control number, the system processes it as I5G4 and only one transaction for each group is allowed.
2. DataInterchange always uses numbers for control number values. Therefore, leading zeros are removed unless:
 - They are needed to satisfy a minimum length requirement.
 - Part of a higher level control number is used to create a lower level control number (for example, I5V in the transaction control number field). The control number generated always has the maximum length as defined by the standard unless it is a transaction control number and a Vn is used.
3. These codes must be entered in uppercase (for example, I5V).

Trading Partner Determination

It is important to understand the search sequence used when an interchange is parsed to determine which trading partner is going to receive this interchange (during the send process), or which trading partner has sent this interchange (during the develope process). DataInterchange uses the following steps to identify the trading partner:

1. Using the qualifier and ID from the interchange, DataInterchange searches the trading partner profile members for a match in the *Interchange qualif* and *Interchange ID* fields.
2. If the interchange ID is 15 or fewer bytes, DataInterchange parses the interchange ID into a 7-byte account number and an 8-byte user ID, then searches the trading partner profile members for a match in the *Account number* and *User ID* fields.
3. If the interchange ID is 16 or fewer bytes, DataInterchange parses the interchange ID into an 8-byte account number and an 8-byte user ID, then searches the trading partner profile members for a match in the *Account number* and *User ID* fields.
4. If the interchange ID is 35 or fewer bytes, DataInterchange parses the interchange ID into a 32-byte account number and a 3-byte user ID, then searches the trading partner profile members for a match in the *Account number* and *User ID* fields.
5. DataInterchange parses the interchange ID into an account number and a user ID separated by either blanks, periods, or slashes, then searches the trading partner profile members for a match in the *Account number* and *User ID* fields.
6. Using the interchange ID, DataInterchange searches the trading partner profile members for a match in the *Contact phone* field.
7. Using the interchange ID, DataInterchange searches TTABLE to determine an account number and user ID, then searches the trading partner profile members for a match in the *Account number* and *User ID* fields.

Note: Some network programs, such as Expedite Base/MVS Version 4 Release 1 and Expedite/MVS Host Version 1 Release 3, also read the TTABLE. However, these network programs may expect different record formats. Although DataInterchange is able to process a TTABLE with either format, the individual network programs require the TTABLE to have the specific format defined by them.

Provide a value in the *Interchange qualif* and *Interchange ID* fields of each trading partner profile member for the best response during any send or receive process.

Note: With 0 enveloping (no interchange, or send usage envelope type = 0, or ISA06 is blank), DataInterchange uses the trading partner nickname.

Handling a Trading Partner in Multiple Environments

You may need to translate and send data to one trading partner from both MVS and CICS. For example, the capability to translate and send from both environments is very helpful when one department processes data in MVS and another in CICS, but both sets of data go to the same trading partner. Another case is when data is processed in CICS, but is switched to MVS in emergency situations. DataInterchange provides a method for performing all functions in either environment for the same trading partner.

When defining the trading partner profile member, you must choose a *primary* and a *secondary* environment. The primary environment is determined by the value in the *Network ID* field of the trading

partner profile member. To use CICS as the primary environment, enter IINCICS in the *Network ID* field. To use MVS as the primary environment, enter one of the following in the *Network ID* field:

- IINB1
- IINB41
- IINB42
- IINR3

The environment you specified in the *Network ID* field becomes the primary environment, the other becomes the secondary environment. For example, if you specified a CICS network ID as the primary environment, MVS becomes the secondary environment.

In the primary environment, you can use all functions as documented in *DataInterchange Programmer's Reference*. All functions can also be performed in the secondary environment, except for the following DataInterchange Utility combination functions:

- PERFORM TRANSLATE AND SEND
- PERFORM ENVELOPE AND SEND
- PERFORM REENVELOPE AND SEND

In the secondary environment, enveloping and sending functions must be split apart when using the DataInterchange Utility. During these combination-send functions, the network IDs to be sent are retrieved from the applicable trading partner profile members, NOT from the requestor profile members. If the combination-send function is attempted in the secondary environment and a trading partner profile member contains a primary environment network ID, that network ID is used. The send will fail because a CICS network program cannot be executed in MVS, and an MVS network program cannot be executed in a CICS environment.

To solve this problem, the translation and send functions must be executed separately. In the following example, the primary environment is MVS, and the secondary environment is CICS. The application data in transient data queue IN01 is destined for trading partners whose trading partner profiles specify IINB1 as the network ID. The file ID override tells DataInterchange to write all envelopes to a temporary storage queue named EDIQDAT. EDIQDAT is the default name of the envelope TS queue supplied with network profile IINCICS. Requestor profile IINCICSREQ has IINCICS as the network ID. This example would be executed in CICS, the secondary environment.

```
PERFORM TRANSLATE AND ENVELOPE
WHERE APPFILE(IN01) APPTYPE(TD) FILEID(EDIQDAT)
PERFORM SEND WHERE REQID(IINCICSREQ)
```

Because the translation and sending functions are split apart, DataInterchange is not concerned with networks affected by the enveloping function. Data is written to TS queue EDIQDAT, and the separate send function instructs DataInterchange to send the data using the IINCICS network profile.

Continuous Receive Facility

The Continuous Receive Facility is a DataInterchange service that works only with Expedite/CICS and MQSeries Queues. You can automate receiving information from the mailbox or MQSeries Queues by defining members of a continuous receive profile to:

- Receive and deenvelope standard data
- Translate the standard data to application format

Communications with Trading Partners

- Automatically start a response application, which receives the application data into a temporary storage queue
- Automatically receive network acknowledgments

The Continuous Receive Facility allows you to implement *event-driven EDI*, which triggers additional processing when a specified event occurs.

Setting Up the Continuous Receive Profile

Add one member to this profile for each unique continuous receive session you want to run. For example, you can add one member for each network mailbox (requestor ID), or one for each transaction type (message user class), or one for a particular trading partner. You can also add members to define different ways of processing the transactions you receive. Processing options include:

- Translating the data and delivering it to a response application
- Saving the untranslated data in the Transaction Store
- Providing the output in C (control) and D (data) records or as raw data
- Starting a response transaction or application after DataInterchange has finished its processing
- Automatically receiving network acknowledgments

To add a member to the Continuous Receive profile, follow these steps:

1. From the Administrator's Menu, select *Profiles*. The Profile Definitions panel (PM01) is displayed.
2. Type **M** in the action column next to CONTRECV, and press Enter.

Log work with Members Print View			
PM01		Profile Definitions	1 to 17 of 17
Action	Profile ID	Profile Description	Log?
—	ACTLOGS	Activity log	N
—	ADAMCTL	User program information	N
—	APPDEFS	Application Definition Profile	N
m	CONTRECV	Continuous Receive Profile	N
—	E	EDIFACT standard envelope data	N
—	I	ICS standard envelope data	N
—	LANGPROF	Language profile	N
—	MQSERIES	MQSeries queue profile	N
—	NETOP	Network operation profile	N
—	NETPROF	Network profile	N
—	REQPROF	Requestor profile	N
—	SECUPROF	Security profile	N
—	SYSPROF	System profile	N
—	T	UN/TDI standard envelope data	N
—	TPPROF	Trading partner profile	Y
—	U	UCS standard envelope data	N
—	X	X12 standard envelope data	N

The Profile Members panel (PM07) is displayed.

3. Type **A** in the action column next to any item, and press Enter.

Add Copy Delete List Print Update View			
PM01		Profile Definitions 1 to 17 of 17	
Action	Profile ID	PM07	Profile Members 1 to 1 of 1
-	ACTLOGS		
-	ADAMCTL		
-	APPDEFS		
m	CONTRECV	Profile ID : CONTRECV Description: Continuous Receive Profile	
-	E		
-	I	Action Key a CONTRECV1	
-	LANGPROF		
-	MQSERIES	MQSeries queue profile	N
-	NETOP	Network operation profile	
-	NETPROF	Network profile	N
-	REQPROF	Requestor profile	N
-	SECUPROF	Security profile	N
-	SYSPROF	System profile	N
-	T	UN/TDI standard envelope data	N
-	TPPROF	Trading partner profile	Y
-	U	UCS standard envelope data	N
-	X	X12 standard envelope data	N

The Add Profile Member panel (PM08) is displayed.

PM08		Add Profile Member		1 to 13 of 24	
Profile ID: CONTRECV					
Fill in the information below and press Enter to save this member. To stop entering members, press Exit or Cancel.					
Continuous Recv ID	_____				
Active	_____				
Requestor ID	_____				
TP nickname	_____				
Message user class	_____				
Translate	_____				
Raw data	_____				
Print name	_____				
Print type	_____				
Exception name . .	_____				
Exception type . .	_____				
Additional records	_____				
Deenvelope only	_____				

PM08		Add Profile Member		13 to 24 of 24	
Profile ID: CONTRECV					
Fill in the information below and press Enter to save this member. To stop entering members, press Exit or Cancel.					
Deenvelope only . .	_____				
Delay FA's	_____				
FA TS queue	_____				
Response name . . .	_____				
Response type . . .	_____				
User field	_____				
Application ID . .	_____				
National Language	_____				
Allow syncpoints	_____				
Allow dup envelope	_____				
Network acks only	_____				
Purge interval . .	_____				
Track SAP Status. .	_____				
Pageable	_____				

Communications with Trading Partners

4. Type the appropriate information in each field. Press F8 (Forward) to scroll to the remaining panels. If you need additional information about the use of a specific field, place the cursor in that field and press F1 (Help).

The following list describes the information that you type in each of the profile member fields.

In this field:

Enter:

Continuous Recv ID

The name you want to use for this profile member.

Active

Y if this member is available for continuous receive processing, or **N** if it is not.

Requestor ID

The requestor profile member that defines the network mailbox to monitor. This requestor member also contains the message user class that is used by default if you omit the message user class from this continuous receive member. If this continuous receive profile member is based off an MQSeries triggered event, please leave this field blank.

TP nickname

The trading partner from which you want to receive transactions. This field is optional.

Message user class

The code for the classes of information to be received. This field is optional.

Translate

Y if the standard data is placed in the Transaction Store and also translated to application format, **N** if it is not.

Raw data

Y if the translated data is to be stored in raw data format, **N** if it is not. This field is optional.

Print name

The temporary storage queue, transient data queue, or VSAM entry sequenced data set to contain messages regarding the activity associated with processing the continuous receive. The default is PRTFILE. You may also specify &UNIQUE which will tell DataInterchange to generate a uniquely named TS queue. The real name of the TS queue DataInterchange creates and stores information into will be passed on to your response application for further processing.

Print type

The code for the type of storage specified in the *Print name* field. Valid values are:

Code	Description
------	-------------

MQ	MQSeries message queue
TD	Transient data queue
TM	Temporary storage queue (main)
TS	Temporary storage queue (auxiliary)
VS	VSAM entry sequenced data set

The default is TS.

Exception name

The temporary storage queue, transient data queue, or VSAM entry sequenced data set to contain translated transactions that cannot be stored in the application file (because it cannot be opened, for example). The default is FFSEXCP. You may also specify &UNIQUE which will tell DataInterchange to generate a uniquely named TS queue. The real name of the TS queue DataInterchange creates and stores information into will be passed on to your response application for further processing.

In this field: Exception type	Enter: The code for the type of storage specified in <i>Exception name</i> . Valid values are:												
	<table> <thead> <tr> <th>Code</th><th>Description</th></tr> </thead> <tbody> <tr> <td>MQ</td><td>MQSeries message queue</td></tr> <tr> <td>TD</td><td>Transient data queue</td></tr> <tr> <td>TM</td><td>Temporary storage queue (main)</td></tr> <tr> <td>TS</td><td>Temporary storage queue (auxiliary)</td></tr> <tr> <td>VS</td><td>VSAM entry sequenced data set</td></tr> </tbody> </table>	Code	Description	MQ	MQSeries message queue	TD	Transient data queue	TM	Temporary storage queue (main)	TS	Temporary storage queue (auxiliary)	VS	VSAM entry sequenced data set
Code	Description												
MQ	MQSeries message queue												
TD	Transient data queue												
TM	Temporary storage queue (main)												
TS	Temporary storage queue (auxiliary)												
VS	VSAM entry sequenced data set												
Additional records	Additional record types you want returned with the translated data. This field is optional. Valid values are:												
	<table> <thead> <tr> <th>Code</th><th>Description</th></tr> </thead> <tbody> <tr> <td>I</td><td>Information records</td></tr> <tr> <td>E</td><td>Envelope (interchange) header records</td></tr> <tr> <td>G</td><td>Group header records</td></tr> <tr> <td>T</td><td>Transaction set header records</td></tr> <tr> <td>Q</td><td>Queuing totals</td></tr> </tbody> </table>	Code	Description	I	Information records	E	Envelope (interchange) header records	G	Group header records	T	Transaction set header records	Q	Queuing totals
Code	Description												
I	Information records												
E	Envelope (interchange) header records												
G	Group header records												
T	Transaction set header records												
Q	Queuing totals												
	See the <i>DataInterchange Installation Guide</i> for descriptions of these records.												
Deenvelope only	Y if the transactions are deenveloped and placed in the Transaction Store but not translated to application format, or N if transactions are deenveloped and translated. This field is optional.												
	This field is ignored if either the <i>Network acks only</i> field or the <i>Translate</i> field is Y.												
Delay FA's	Y if enveloping of functional acknowledgments, if it is delayed, or N if it is not delayed. This field is optional.												
FA TS queue	The temporary storage queue for holding enveloped functional acknowledgments. This field is optional. This field is ignored if the <i>Delay FAs</i> field is Y. You may also specify &UNIQUE which will tell DataInterchange to generate a uniquely named TS queue. The real name of the TS queue DataInterchange creates and stores information into will be passed on to your response application for further processing.												
Response name	The CICS transaction or program to which DataInterchange gives control after processing. This field is optional.												
Response type	TX if the response name is for a transaction or PG for a program. This field is optional.												
User field	A 16-byte area set aside for any purpose you choose. This field is optional. This field will be available to your response program when it receives control from DataInterchange.												
Application ID	The ID of the application that initialized DataInterchange. This field is optional.												
National Language	The national language code for the session. Release 4 and higher releases are only available in US English. The language code is ENU.												
Allow syncpoints	Y if DataInterchange is allowed to issue CICS SYNCPOINT commands when envelope processing completes, or N if it is not allowed. This field is optional.												

Communications with Trading Partners

In this field:	Enter:
Allow dup envelope	Y if duplicate envelopes should be processed during deenveloping, or N if they should not. This field is optional.
Network acks only	Y if the continuous receive request is for network acknowledgment only, or N if it is not. This field is optional.
Purge interval	The number of days a transaction is kept in the Transaction Store before it is marked for purging. This field is optional. The default is 30 days.
Track SAP Status	Y to turn SAP status tracking on, or N to not turn status tracking on.
Pageable	Optional. Enter a Y (for yes) or an N (for no) to indicate whether or not Pageable Translation should be used. If this value is not supplied, the default value of N is presumed. When the amount of envelope and application data together exceeds 28 MB, Pageable Translation will begin paging the excess to an auxiliary storage TS queue. Pageable Translation is designed to better utilize virtual storage.

5. Press Enter. The Add Profile Member panel (PM08) is redisplayed. You can add as many members as necessary for your organization. Press F3 (Exit) when you are done adding members.

For information about starting or stopping continuous receive requests, continuous receive cleanup, or MQSeries trigger event setup, see the *DataInterchange Programmer's Reference*.

Requesting, Sending, and Receiving Functional Acknowledgments

For some transactions you send, you may want the receiver to reply, acknowledging receipt of the transaction. The acknowledgment could convey acceptance or rejection. Such a reply is called a *functional acknowledgment*. The contents of the reply are defined by the X12 997 or UCS 999 transaction set, or by the EDIFACT CONTRL message. This discussion, therefore, applies only for those who use the X12, UCS, or EDIFACT standards.

To exchange functional acknowledgments, both the requestor and the sender must take certain steps. These steps are described in the following sections, along with explanations of how the exchange occurs.

Requesting Functional Acknowledgments

To request functional acknowledgments, follow these steps:

1. Tell your trading partner the type of transaction for which you want to receive an acknowledgment.
2. In the *send* usage for the trading partner transaction, set the *Acknowledgment expected* field to **Y**. This tells DataInterchange to set "pending functional acknowledgment" status for these transactions when they are sent.

Sending Functional Acknowledgments

Note: A mapping must exist for a transaction that is deenveloped before a functional acknowledgment will be created.

To send functional acknowledgments, follow these steps:

1. In the *receive* usage for the trading partner transaction to be acknowledged, enter the *Acknowledgment type*. The acknowledgment type is either 997, 997V35, 999, CONTRL, or CONTV21. This informs the translator to build the functional acknowledgment, and either put it in the Transaction Store and queue it for sending or only put it in the Transaction Store (see "Delayed

Functional Acknowledgments” on page 5-50). By default, the name of the queue is QDATA (for 997), QDATAU (for 999), or QDATAE (for CONTRL). You may change the name by updating the network profile member in which it appears or by including the FUNACKFILE keyword in the deenvolving request. The FUNACKFILE keyword overrides the name of the transaction data queue in the network profile. See “Transaction data queue” on page 5-11 and the description of the FUNACKFILE keyword in the *DataInterchange Programmer’s Reference*.

- To generate a 997 functional acknowledgment, enter either 997V35 for Version 3 Release 5 and later, or 997 for the version and release prior to Version 3 Release 5.
- To generate a CONTRL functional acknowledgment, enter either CONTV21 for Version 2 Release 1 and later, or CONTRL for the version and release prior to Version 2 Release 1.

The JCL for the job that receives and translates the transaction to be acknowledged must contain a DD statement with the name of the file, as in the following example:

```
//QDATA DD DSN=IIN.TRANS,DISP=MOD
```

2. Use the enveloping options file (FAENV) if you need to override values placed in the envelope segments. See the *DataInterchange Programmer’s Reference* for details.
3. Use the Inbound envelope on FA flag to indicate the use of the inbound envelope information for enveloping the Functional Acknowledgment. See the description of Inbound envelope used on FA on page 9-92 for more information.
4. Transaction Store pairs the functional acknowledgment with the transaction that caused its generation. You can view or print the acknowledgment by choosing the *Acknowledgment* or *Print* action from a Transaction Store Facility panel.
5. Send the acknowledgment, along with any other transactions in the file, to the trading partner. DataInterchange does not automatically send the file of acknowledgments. You must issue the send command separately.

Receiving Functional Acknowledgments

The trading partner who requested the acknowledgment receives the 997, 999, or CONTRL, along with other transactions that may be in the same envelope. The translator translates all the transactions and puts them in the application file. (See the note below for an exception.) It also logs the appropriate transaction images, depending on user choices and on any errors that may occur. Transaction Store pairs the functional acknowledgment with the transaction that requested it and updates the status of that transaction. The partner who requested the acknowledgment may view or print this status information by choosing the *Acknowledgment* or *Print* action from a Transaction Store panel.

Notes:

1. DataInterchange uses the functional acknowledgment to update the status information. It translates the functional acknowledgment and places it in the application file only if you have generated a mapping to receive the functional acknowledgment.
2. DataInterchange interrogates the version and release in the UNH segment of the received EDIFACT CONTRL functional acknowledgment to determine the format of the received CONTRL message. The version and release must be equal to or greater than Version 2 Release 1 in order for DataInterchange to accurately interpret the CONTRL message and reconcile it with messages awaiting the acknowledgments. Any version prior to 2 is interpreted using the older version of the CONTRL message. You should notify all your trading partners who send the new format CONTRL message that the UNH must contain the correct version and release.

Delayed Functional Acknowledgments

If you do not want to queue functional acknowledgments immediately, use the FADELAY keyword on the deenveloping request. The values for this keyword are Y and N, where:

- Y Indicates that functional acknowledgments are to be put in the Transaction Store.
- N Indicates that functional acknowledgments are to be put in the Transaction Store, and also enveloped to one of the following:
 1. The file specified by the FUNACKFILE keyword, if present
 2. The transaction data queue name from the network profile (second choice)
 3. QDATA, QDATAU, or QDATAE (depending on the envelope type) by default

Note: For DataInterchange for CICS, FUNACKFILE is a temporary storage queue. You can also delay enveloping of functional acknowledgments using the *Delay FA's* field in the continuous receive profile. See "Setting Up the Continuous Receive Profile" on page 5-44 for details.

Interchange Acknowledgments (TA1 Segment)

DataInterchange ignores the TA1 segment within an X12 envelope being deenveloped and does not create the TA1 segment in X12 envelopes when creating the envelope.

Note: We recommend you use functional acknowledgments rather than interchange acknowledgments to ensure your trading partners get your transactions. If you receive X12 envelopes with TA1 segments and no groups or transactions, you may be left with orphan interchange envelopes in the Transaction Store. This condition only happens if envelope level recovery is used. To remove orphan interchange envelopes in the Transaction Store, use the following SQL statement:

```
DELETE FROM EDIENU31.EDITSEV WHERE SUBSTR(TRLIMAGE,1,6)='IEA*0*';
```

Chapter 6. Customizing EDI and Envelope Standards

Transactions/Messages, Segments, and Data Elements	6-1
Envelopes	6-3
Customizing EDI Standards	6-4
Copying an EDI Standard	6-7
Adding a Data Element or Composite Element	6-9
Updating a Composite Element	6-11
Adding a Segment	6-13
Adding a Transaction Set	6-15
Determining Loop IDs	6-18
Data Types for EDI Standard Data Elements	6-20
Adding an EDI Standard	6-22
Viewing Standards, Transaction Sets, Segments, Composite Elements, and Data Elements	6-23
Customizing Envelope Standards	6-26
Providing Envelope Data	6-26
Updating an Envelope Standard	6-27
Updating Envelope Segments	6-28
Using Data Types in Envelope Standards	6-29
Envelope Profiles	6-30
Adding a Member to an Envelope Profile	6-30
Data Types for Envelope Standards	6-32
EDIFACT Envelope Data	6-35
ICS Envelope Data	6-42
UNTDI Envelope Data	6-46
UCS Envelope Data	6-50
X12 Envelope Data	6-54
Building the Enveloping Segments	6-60

Chapter 6. Customizing EDI and Envelope Standards

DataInterchange separates the management of data elements and segments that comprise the EDI envelopes from the data elements and segments that comprise transaction sets or messages.

DataInterchange refers to these as *envelope standards* and *EDI standards*, respectively.

When you decide to exchange information through electronic data interchange (EDI), whether at your request or another company's, you must select an EDI standard to define the information you want to send or receive. Ideally, you and your trading partner can agree on an EDI standard that requires little or no customization to meet your needs, but this is not always possible.

If you are driving the use of EDI, try to fit your business needs into an existing standard. This makes it easier on both you and your trading partner, which makes your EDI program more successful. If another company requested EDI, that company should provide you with the standard it wants to use, the parts of the standard to use, and any other appropriate information.

DataInterchange gives you the ability to do work with EDI standards in a variety of ways. You can:

- add data elements, composite data elements, and transactions
- view data elements, composite data elements, and transactions
- update data elements, composite data elements, and transactions

In addition, standards also define lists of acceptable values for data elements. DataInterchange stores these lists as validation tables. For more information, see Chapter 8, "Translation and Validation Tables."

Transactions/Messages, Segments, and Data Elements

Transaction and *message* are general names for a business document such as an invoice or a purchase order. When translated to EDI standard format, the document becomes a *transaction set* or *message*. As with a conventional printed document, most transaction sets or messages have three general areas:

- The *heading area* contains information about the whole document, such as the company name and address.
- The *detail area* contains the transaction details, such as quantities ordered and prices.
- The *summary area* contains more information about the transaction as a whole, such as the amount due.

In general, each line of a business document corresponds to a *segment* in the EDI transaction set or message. The units of data that make up a segment are *data elements*. A line item on an invoice is an example of a segment, while item numbers and unit prices are examples of data elements.

The characteristics of segments and data elements are summarized below. See the documentation for the standard you are using for complete details.

Customizing EDI and Envelope Standards

Segments:

- Begin with a segment identifier assigned by the standard.
- Are the following:
 - mandatory* (always present)
 - conditional* (requirement depends on the nature of the transaction)
 - optional* (your choice)
 - floating* (may appear anywhere in the transaction)
- Appear in a fixed sequence for a given transaction, floating segments being the exception.
- May repeat within the transaction. The *maximum use* limits the number of times a segment can repeat.
- May form groups referred to as *loops* and identified by a *loop ID*. The entire loop may be repeatable. *Loop repeat* limits the number of times the loop can appear in succession. The segments making up a name and address are an example of a loop.
- End with a segment terminator such as the new line character.

Data elements:

- Appear in a sequence specified by the standard.
- Are separated by a delimiting character such as an asterisk.
- Are either *mandatory* (always present), *conditional* (requirement depends on presence of other data elements), or *optional* (your choice).
- Have a minimum and a maximum length.
- Must be of a *data type* prescribed by the standard, such as date, time, and alphanumeric.
- If they are identifiers (data type ID), contain one of the codes prescribed by the standard. An example is the data element for unit of measure. The standard specifies the acceptable codes, such as CA for case. You can customize the list of acceptable codes.

The following illustration shows a typical transaction in paper format and the EDI standard format of one line item.

Paper Format

WHOLESALE ANGLING SUPPLIES 101 Main Street Blairsville, GA 30512		INVOICE No. 414		
Bill to: Blue Ridge Tackle Shop 298 Lakeside Drive Blue Ridge, GA. 30442		Ship to: Blue Ridge Tackle Shop 298 Lakeside Drive Blue Ridge, GA. 30442		
Quantity	Item No.	Description	U/M	Price
1	750	FLY ROD	EA	79.95
2	850	FLY REEL	EA	29.95
Date ordered: 7/11/88		Pay this Amount		139.85

EDI Format of One Segment

IT1	**	2	*	EA	*	29.9500	**	VC	*	850	N/L
-----	----	---	---	----	---	---------	----	----	---	-----	-----

Segment ID

Data Element

Data Element Separator

Segment Terminator

Figure 6-1. Paper Format and EDI Standard Format of a Transaction

Note: In the EDI format, an asterisk indicates a data element separator. A double asterisk indicates that a data element in the standard is not used in the paper document.

Envelopes

Just as a postal envelope may contain more than one document, an electronic transmission may contain more than one transaction set or message. It may even contain transaction sets or messages of different types, such as purchase orders and invoices. To separate the types, the envelope standards define *header* and *trailer* segments that mark the beginning and ending of each transaction set or message, and of each group of the same type. These header and trailer segment combinations are the envelopes for electronic mail.

As Figure 6-2 on page 6-4 shows, the standards define three levels of enveloping:

- The innermost level encloses a transaction set or message. The beginning and ending segments are the transaction set header and trailer.
- The next level of enveloping is for a group of transaction sets or messages of the same type. The beginning and ending segments are the *functional group* header and trailer.

Customizing EDI Standards

- The outermost level of enveloping identifies the trading partner that is to receive the mail. The beginning and ending segments are the *interchange control* header and trailer.

The transmission system adds an additional level of enveloping to control the communication session. This envelope is not part of the standards and does not appear in the illustration.

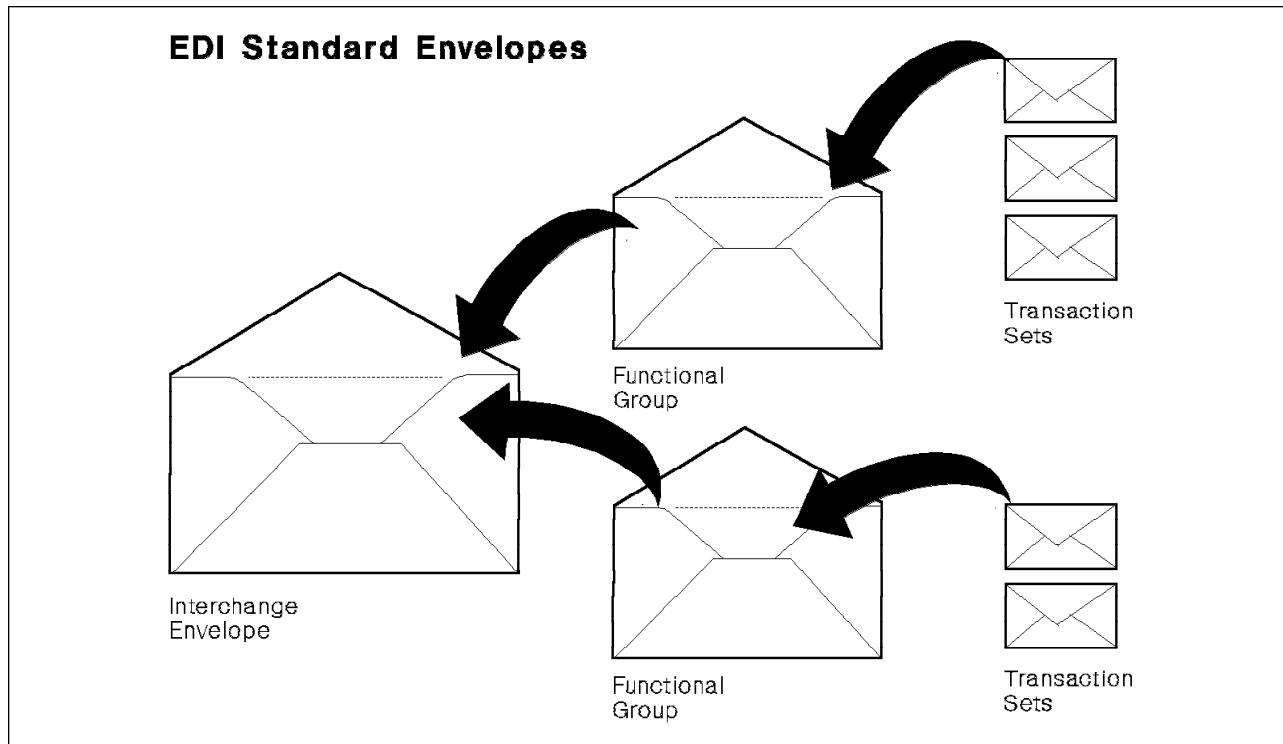


Figure 6-2. Levels of Enveloping

It is possible during translation to cluster transactions. The Transaction Store Facility and DataInterchange Utility ensure that any action that takes place against any member of the cluster also takes place against all members of the cluster. If any member is enveloped, all members are enveloped.

Customizing EDI Standards

DataInterchange lets you tailor your EDI standard transaction sets to match your trading partner's transaction sets or the recommendations of a specific industrial sector such as insurance. As defined by their issuers, standards are flexible enough to fit a wide range of businesses. Some of the optional segments and data elements will not apply to everyone. You and your trading partners must, therefore, agree on the handling of these optional units. After agreeing on the content and format of the transactions, you can customize the standard to fit your needs.

When you select *EDI standards* from the Administrator's Menu (MP01), the system displays the EDI Standards panel (SC01), which lists the standards available on your system.

Note: If the Add EDI Standard panel (SC02) is displayed, use the apply standard program to select the standards for your organization and make them available for use. For more information, see Chapter 3, "Importing and Requesting EDI Standards Electronically."

You can work with an entire standard, or use the commands on the action bar to work with the individual transaction sets, segments, or data elements in the standard. At each level, you can add, copy, delete, print, update, or view the items. You can also get a list of trading partner transactions that have been

mapped to the transaction sets, segments, and data elements of an EDI standard. Table 6-1 on page 6-5 summarizes the task options for EDI and envelope standards.

Table 6-1 (Page 1 of 2). Summary of Tasks for EDI and Envelope Standards

Action	EDI Standards (SC01) Envelope Stds (SC16)	Transactions (SC17)	Segment Directory (SC11)	Data Element Dictionary (SC06)
Add	<ul style="list-style-type: none"> Adds a standard to the database. Defines the data elements in the data element dictionary. Forms segments for EDI and envelope standards. Forms transaction sets for EDI standards. 	Selects segments in the transaction set.	Selects data elements in the segment.	Defines a new data element.
Copy	Copies an entire standard (data elements, segments, and transaction sets) to another standard ID.	Copies a standard transaction to another transaction ID.	Copies the data elements of a segment to another segment ID.	Copies a data element to another data element ID.
Delete	Deletes an entire standard (data elements, segments, and transaction sets). You cannot delete an envelope standard used by an EDI standard. The system warns you if an EDI standard is used in a trading partner transaction.	Deletes a standard transaction. The system warns you if the standard transaction is used in a trading partner transaction.	Deletes a segment. You cannot delete a segment used in a standard transaction.	Deletes a data element. You cannot delete a data element used in a segment.
Generate	<p>Applies to envelope standards only and tells DataInterchange to create a control string for the envelope definition. A control string is a compact set of instructions describing the envelope standard to the DataInterchange translator. Generating a control string is optional, but if done, a large number of database accesses are eliminated each time you start up a translation session. In order to take advantage of this performance enhancement, anytime anything in the envelope definition changes, the Generate should be issued again to rebuild the control string and incorporate the changes made.</p>			
Print	Prints: <ul style="list-style-type: none"> A description of the standard The envelope type The segment IDs and data elements for the headers and trailers A list of the transaction sets defined by the standard 	Prints a listing of the transaction set. If entered on the command line with no transaction ID, prints all transaction sets.	Prints a listing of the segment. If entered on the command line with no segment ID, prints the segment directory.	Prints a listing of the data element. If entered on the command line with no data element ID, prints the data element dictionary.

Customizing EDI Standards

Table 6-1 (Page 2 of 2). Summary of Tasks for EDI and Envelope Standards

Action	EDI Standards (SC01) Envelope Stds (SC16)	Transactions (SC17)	Segment Directory (SC11)	Data Element Dictionary (SC06)
Update	For EDI standards, changes descriptive information and envelope type. For envelope standards, changes descriptive information, delimiters, and header and trailer segment IDs.	Inserts, moves, copies, and deletes segments in the standard transaction. The system warns you if the standard transaction is used for mapping.	Inserts, moves, copies, and deletes data elements in the segment. The system warns you if the segment is used in mapping.	Changes the definition of the data element, including data type, length, and validation table name. The system warns you if the data element is used in mapping.
View	For EDI standards, shows descriptive information and the envelope type. For envelope standards, shows descriptive information and delimiters.	Shows the segments that make up the standard transaction.	Shows the data elements that make up the segment.	Shows the definition of the data element, including data type, length, and validation table name.
Where used	Does not apply to EDI and envelope standards.	Shows the trading partner transactions that are mapped to the standard transaction.	For EDI standards, shows the standard transactions in which the segment appears. For envelope standards, does not apply.	Shows the segments in which the data element appears.
cOmposites	Displays the Composite Element Dictionary panel (SC30).			

Table 6-2 on page 6-7 describes the actions associated with modifying transaction sets, segments, and elements. Modifying includes adding, copying, or updating.

Table 6-2. Tasks for Modifying Transaction Sets, Segments, and Elements

Action	Add Transaction (SC18) Copy Transaction (SC19) Update Transaction (SC20) Update Composite Element (SC32)	Add Segment (SC12) Copy Segment (SC13) Update Segment (SC14)
Block	Marks one or more lines to move or copy.	Marks one or more lines to move or copy.
Copy	Copies a marked block to the line below the one on which you enter the <i>Copy</i> action.	Copies a marked block to the line below the one on which you enter the <i>Copy</i> action.
Delete	Deletes a segment.	Deletes a data element.
Get descriptions	Displays the segment descriptions for lines that contain a segment ID. You can enter this action on any line.	Displays the data element descriptions for lines that contain a data element ID. You can enter this action on any line.
Insert	Inserts a line for you to add segment information in the line below the one on which you enter the <i>Insert</i> action.	Inserts a line for you to add data element information in the line below the one on which you enter the <i>Insert</i> action.
Move	Moves a marked block to the line below the one on which you enter the <i>Move</i> action and erases the marked block from the old location.	Moves a marked block to the line below the one on which you enter the <i>Move</i> action and erases the marked block from the old location.
Segments	Displays the Segment Directory panel (SC11).	
Elements	Displays the Data Element Dictionary panel (SC06).	

Copying an EDI Standard

When you customize an EDI standard, you should make a copy of that standard first, or export the standard. For more information, see Chapter 11, “Exporting and Importing Transactions.” By customizing a copy of the standard, you have the original as a backup or as a base for creating other customized standards.

To copy an EDI standard, follow these steps:

1. From the Administrator’s Menu (MP01), select *EDI standards*. The EDI Standards panel (SC01) is displayed.
2. Type **c** in the action column next to the standard you want to copy, and press Enter.

Add Copy Delete List Print Update View Elements cOmposites Segments Transactions				
SC01		EDI Standards		1 to 14 of 29
A	Standard ID	Description	Vers	Rel
-	AARV3R4	Rail Carrier Industry Version/Release 003040RAIL	03	40
-	CDTEST	UN/EDIFACT Standard (1995 Release 2) Draft 95B	95	02
-	DIUSTD	DIUSERTOADF	01	01
-	EDI891	UN/EDIFACT Standard (1989 Release 1)	89	01
C	EDI902	UN/EDIFACT Standard (1990 Release 2)	90	02
-	EDI94B	UN/EDIFACT Standard (1994 Release 2) Draft 94B	94	02
-	EDI95B	UN/EDIFACT Standard (1995 Release 2) Draft 95B	95	02
-	EXPANDE	Expanded DE sequence number test	01	01
-	E901EI00	UN/EDIFACT (90/1) for Electric Industry Partners	01	01
-	FAORECV	AFTSUFAO	01	01
-	ODEV3	ODETTE Standard (Version 3)	03	00
-	SAMPLE	X12 Sample for the Tutorial		
-	VICSTEST	VICS EDI STANDARDS (Version 3, Release 3)	03	30
-	VICSTST2	VICS EDI STANDARDS (Version 3, Release 3)	03	30
Command ==>				
Enter Tso F1=Help F3=Exit F8=Fwd F9=Retrieve				
F12=Cancel F13=Keys help				

The Copy EDI Standard panel (SC03) is displayed.

- In the **To standard ID** field, type the name you want to use for the copy, for example, **MYEDI902**. If you want to change the description, version, release, or envelope type, press Tab to move the cursor to the appropriate field, then type the new information.

SC03	Copy EDI Standard
From standard ID . . :	EDI902
To standard ID . . .	myedi902
Description	UN/EDIFACT Standard (1990 Release 2) _____
Version	90
Release	02
Envelope type . . .	E

- Press Enter. DataInterchange copies the standard to the name you specified, then redisplay the EDI Standards panel (SC01) with the new standard included in the list.
- Press F3 (Exit) to return to the Administrator's Menu (MP01).

Adding a Data Element or Composite Element

To add a data element or composite element to an EDI standard, follow these steps:

1. From the Administrator's Menu (MP01), select **EDI standards**. The EDI Standards panel (SC01) is displayed.
2. Type **e** in the action column next to the standard to which you want to add the data element or composite element, and press Enter.

```

Add Copy Delete List Print Update View Elements cOmposites Segments
Transactions
-----
SC01                                EDI Standards                                1 to 14 of 29

A Standard ID Description                                Vers Rel
- AARV3R4 Rail Carrier Industry Version/Release 003040RAIL 03 40
- CDTEST UN/EDIFACT Standard (1995 Release 2) Draft 95B 95 02
- DIUSTD DIUSERTOADF 01 01
- EDI891 UN/EDIFACT Standard (1989 Release 1) 89 01
e EDI902 UN/EDIFACT Standard (1990 Release 2) 90 02
- EDI94B UN/EDIFACT Standard (1994 Release 2) Draft 94B 94 02
- EDI95B UN/EDIFACT Standard (1995 Release 2) Draft 95B 95 02
- EXPANDDE Expanded DE sequence number test 01 01
- E901EI00 UN/EDIFACT (90/1) for Electric Industry Partners 01 01
- FAORECV AFTSUFAO 01 01
- ODEV3 ODETTE Standard (Version 3) 03 00
- SAMPLE X12 Sample for the Tutorial
- VICSTEST VICS EDI STANDARDS (Version 3, Release 3) 03 30
- VICSTST2 VICS EDI STANDARDS (Version 3, Release 3) 03 30

Command ==>
Enter Tso F1=Help F3=Exit F8=Fwd F9=Retrieve
F12=Cancel F13=Keys help

```

The Data Element Dictionary panel (SC06) is displayed.

3. Type **a** in the action column next to any data element, and press Enter.

```

Add Copy Delete List Print Update View Where used
-----
SC06                                Data Element Dictionary                                1 to 12 of 350

Standard ID . . . . . : EDI902
Standard description : UN/EDIFACT Standard (1990 Release 2)

A Elem ID Type Description
a C001 CD Entry Type
- C002 CD Document/Message Name
- C003 CD Additional Entry Details
- C004 CD Date
- C006 CD Treaty
- C009 CD Total of open cash claims
- C011 CD Type of Participation
- C012 CD Payment Terms Date
- C013 CD Transferred Amount
- C014 CD Reinsurance Current Account Period
- C016 CD Underwriting Condition
- C017 CD Reinsurance Account Balance
The scrollable list does not include all data elements

```

The Add Data Element panel (SC07) is displayed.

Customizing EDI Standards

The **Standard ID** and **Standard description** fields indicate which standard you are adding the data element to.

SC07	Add Data Element
Standard ID	: EDI902
Standard description . . .	: UN/EDIFACT Standard (1990 Release 2)
Data element ID . .	_____
Description	_____
Data type	__
Minimum length . . .	__
Maximum length . . .	__
Validation table . .	_____

4. Complete the fields as follows:

In this field:

Data element ID

Description

Data type

Minimum length

Maximum length

Validation table

Enter:

The name of the new data element or composite element.

A brief description of the data element. If this is a FIXED standard (envelope type of F) and there is an application data format which is the basis for the standard, then the data element description should be the name of the FIELD within the application data format that corresponds to this data element.

Note: If the name is not the same, then Fixed-to-Fixed may not give the desired results, or generation of a Fixed-to-Fixed mapping will generate an error message.

The data type for the data element. Data types for EDI standards are described in Table 6-3 on page 6-21. The data types allowed for FIXED standards (which are standards defined with an envelope type of F) are allowed to have the same data types as an application data format. The application data format data types are defined in Table 7-2 on page 7-17.

The minimum number of characters permitted. For data type R, do not count the sign or decimal point. For data type N, do not count the sign.

The maximum number of characters permitted. For data type R, do not count the sign or decimal point. For data type N, do not count the sign.

If the data type is ID, the name of the validation table that contains the acceptable values for the data element. The table is used only if the transaction mapping refers to it. For more information, see Chapter 8, "Translation and Validation Tables."

5. After entering the appropriate information, press Enter to save the data element. The Add Data Elements panel (SC07) is redisplayed with the new data element included in the list. To add another data element, repeat steps 3 through 5.

6. Press F3 (Exit) twice to return to the Administrator's Menu (MP01).

After adding a composite data element definition, you must update the composite data element to define the components for the composite. See "Updating a Composite Element" on page 6-11 for more information.

Updating a Composite Element

To update a composite element to an EDI standard, follow these steps:

1. From the Administrator's Menu (MP01), select **EDI standards**. The EDI Standards panel (SC01) is displayed.
2. Type **o** in the action column next to the standard of the composite element you want to work with, and press Enter.

```

Add Copy Delete List Print Update View Elements cOmposites Segments
Transactions
-----
SC01                                EDI Standards                                1 to 14 of 29

A Standard ID Description                                Vers Rel
- AARV3R4 Rail Carrier Industry Version/Release 003040RAIL 03 40
- CDTEST UN/EDIFACT Standard (1995 Release 2) Draft 95B 95 02
- DIUSTD DIUSERTOADF 01 01
- EDI891 UN/EDIFACT Standard (1989 Release 1) 89 01
- EDI902 UN/EDIFACT Standard (1990 Release 2) 90 02
o EDI94B UN/EDIFACT Standard (1994 Release 2) Draft 94B 94 02
- EDI95B UN/EDIFACT Standard (1995 Release 2) Draft 95B 95 02
- EXPANDDE Expanded DE sequence number test 01 01
- E901EI00 UN/EDIFACT (90/1) for Electric Industry Partners 01 01
- FAORECV AFTSUFAO 01 01
- ODEV3 ODETTE Standard (Version 3) 03 00
- SAMPLE X12 Sample for the Tutorial
- VICSTEST VICS EDI STANDARDS (Version 3, Release 3) 03 30
- VICSTST2 VICS EDI STANDARDS (Version 3, Release 3) 03 30

Command ==>
Enter Tso F1=Help F3=Exit F8=Fwd F9=Retrieve
F12=Cancel F13=Keys help

```

The Composite Element Dictionary panel (SC30) is displayed.

3. Type **u** in the action column next to the composite element you want to update, and press Enter.

```

List Print Update View Where used
-----
SC30                                Composite Element Dictionary                                1 to 12 of 59

Standard ID . . . . . : EDI94B
Standard description : UN/EDIFACT Standard (1994 Release 2) Draft 94B

A Comp ID Type Description
u C002 CD Document/message name
- C040 CD Carrier
- C056 CD Department or employee details
- C058 CD Name and address
- C059 CD Street
- C076 CD Communication contact
- C078 CD Account identification
- C080 CD Party name
- C082 CD Party identification details
- C088 CD Institution identification
- C100 CD Terms of delivery or transport
- C107 CD Text reference

Command ==>
Enter Tso F1=Help F3=Exit F8=Fwd F9=Retrieve
F12=Cancel F13=Keys help

```

Customizing EDI Standards

The Update Composite Element panel (SC32) is displayed.

Block Copy Delete Get descriptions Insert Move Elements

SC32 Update Composite Element 1 to 4 of 4

Standard ID : EDI94B
Standard description : UN/EDIFACT Standard (1994 Release 2) Draft 94B
Composite ID : C002
Composite description : Document/message name

A	Ref	Element	Req	Rel	Related Element	Description
	Num	ID	Des	Def	Reference Numbers	
-	1	1001	C	-		Document/message name, code
-	2	1131	C	-		Code list qualifier
-	3	3055	C	-		Code list responsible agency
-	4	1000	C	-		Document/message name

Command ==>
Enter Tso F1=Help F3=Exit F9=Retrieve
F12=Cancel F13=Keys help

4. Complete the fields as follows:

In this field:

Element ID

Req Des

Rel Def

Enter or Change:

The ID of a data element defined in the EDI standard.

The code for the required use of the data element. Use **M** for mandatory, **C** for conditional, or **O** for optional.

If you entered **C** in this column, you must also fill in the Related Definition Code and the Related Element Reference Numbers of the data elements that are included in the relationship. The valid codes are:

Code Description

P If one related element is present, all must be present (paired).

R At least one related element must be present (required).

E Only one related element can be present (exclusive).

C If the first related element is present, all others must be present (conditional).

L If the first related element is present, at least one other must be present (conditional paired).

Related Element

Reference Numbers

The numbers, from the **Ref Num** column, of the related data elements.

Leave this field blank if the **Rel Def** field is blank.

5. After entering the appropriate information, press Enter to save the new composite element. The Composite Element Dictionary panel (SC30) is displayed with the list of composite elements included for this standard.

Function keys for panel SC32 are described on page 2-7. The actions for modifying transaction sets, segments, and elements are described in Table 6-2 on page 6-7. If you fill all the available lines and press Enter, another panel of lines is displayed. If you do not need the extra lines, go to the next step.

6. To update another composite element, repeat steps 3 through 5. To add a composite element, follow the steps in “Adding a Data Element or Composite Element” on page 6-9.
7. Press F3 (Exit) twice to return to the Administrator’s Menu (MP01).

Adding a Segment

Before you add a segment, ensure that all the data elements you want to use in the segment are defined to DataInterchange.

To add a segment to an EDI standard, follow these steps:

1. From the Administrator’s Menu (MP01), select **EDI standards**. The EDI Standards panel (SC01) is displayed.
2. Type **s** in the action column next to the standard you want to add the segment to, and press Enter.

Add Copy Delete List Print Update View Elements cOmposites Segments Transactions				
SC01		EDI Standards		1 to 14 of 29
A	Standard ID	Description	Vers	Rel
—	AARV3R4	Rail Carrier Industry Version/Release 003040RAIL	03	40
—	CDTEST	UN/EDIFACT Standard (1995 Release 2) Draft 95B	95	02
—	DIUSTD	DIUSERTOADF	01	01
—	EDI891	UN/EDIFACT Standard (1989 Release 1)	89	01
s	MYEDI902	UN/EDIFACT Standard (1990 Release 2)	90	02
—	EDI94B	UN/EDIFACT Standard (1994 Release 2) Draft 94B	94	02
—	EDI95B	UN/EDIFACT Standard (1995 Release 2) Draft 95B	95	02
—	EXPANDDE	Expanded DE sequence number test	01	01
—	E901EI00	UN/EDIFACT (90/1) for Electric Industry Partners	01	01
—	FAORECV	AFTSUFAO	01	01
—	ODEV3	ODETTE Standard (Version 3)	03	00
—	SAMPLE	X12 Sample for the Tutorial		
—	VICSTEST	VICS EDI STANDARDS (Version 3, Release 3)	03	30
—	VICSTST2	VICS EDI STANDARDS (Version 3, Release 3)	03	30
Command ==>				
Enter Tso F1=Help F3=Exit F8=Fwd F9=Retrieve				
F12=Cancel F13=Keys help				

The Segment Directory panel (SC11) is displayed.

3. Type **a** in the action column next to any segment, and press Enter.

Add Copy Delete List Print Update View Where used		
SC11	Segment Directory	1 to 12 of 114
Standard ID : MYEDI902		
Standard description : UN/EDIFACT Standard (1990 Release 2)		
A	Seg ID	Description
a	ACA	Alternative Currency Amount
-	ACT	Alternative Currency Total Amount
-	AJT	Monetary Amount Adjustment Details
-	ALC	Allowance Or Charge
-	ALI	Additional Information
-	API	Additional Price Information
-	AUT	Authentication Result
-	BGM	Beginning of Message
-	BUS	Business Function
-	CAE	Reinsurance Current Account Entry
-	CAI	Reinsurance Current Account Identification
-	CBD	Cost Breakdown

The Add Segment panel (SC12) is displayed. The **Standard ID** and **Standard description** fields indicate the standard to which you are adding the segment.

Block Copy Delete Get descriptions Insert Move Elements							
SC12	Add Segment						1 to 9 of 20
Standard ID : MYEDI902							
Standard description : UN/EDIFACT Standard (1990 Release 2)							
Segment ID : _____							
Segment description . . : _____							
A	Ref Num	Element ID	Req Des	Rel Def	Related Element Reference Numbers	Description	
-	1	_____	-	-	_____	_____	
-	2	_____	-	-	_____	_____	
-	3	_____	-	-	_____	_____	
-	4	_____	-	-	_____	_____	
-	5	_____	-	-	_____	_____	
-	6	_____	-	-	_____	_____	
-	7	_____	-	-	_____	_____	
-	8	_____	-	-	_____	_____	
-	9	_____	-	-	_____	_____	

4. Complete the fields as follows:

In this field:

Segment ID

Segment description

Enter:

The name of the new segment.

A brief description of the segment. If this is a FIXED standard (envelope type of F) and there is an application data format which is the basis for the standard, then the segment description should be the name of the STRUCTURE within the application data format that corresponds to this segment.

Note: If the name is not the same, then Fixed-to-Fixed may not give the desired results, or generation of a Fixed-to-Fixed mapping will generate an error message.

Element ID

The ID of a data element defined in the EDI standard.

Req Des

The code for the required use of the data element. Use **M** for mandatory, **C** for conditional, or **O** for optional.

In this field:

Rel Def

Enter:

If you entered **C** in the *Req Des* column, the code defining the relationship between the data element in the *Element ID* column and the data elements in the *Related Element Reference Numbers* columns:

Code Description

P	If one related element is present, all must be present (paired).
R	At least one related element must be present (required).
E	Only one related element can be present (exclusive).
C	If the first related element is present, all others must be present (conditional).
L	If the first related element is present, at least one other must be present (conditional paired)

Related Element

The numbers, from the *Ref Num* column, of the related data elements.

Reference Numbers

Leave this field blank if the **Rel Def** field is blank.

5. After entering the appropriate information, press Enter to save the new segment. The Segment Directory panel (SC11) is redisplayed with the new segment included in the list.

Function keys are described on page 2-7. The actions for transactions, segments, and elements are described in Table 6-2 on page 6-7. If you fill all the available lines and press Enter, another panel of lines is displayed. If you do not need the extra lines, go to the next step.

6. To add another segment, repeat steps 3 through 5.
7. Press F3 (Exit) twice to return to the Administrator's Menu (MP01).

Adding a Transaction Set

Before you add a transaction set, ensure that all the segments you want to use in the transaction set are defined to DataInterchange. If a transaction set exists that contains segments similar to what you need, it may be easier to copy the transaction set and change it to meet your needs.

To add a transaction set to an EDI standard, follow these steps:

1. From the Administrator's Menu (MP01), select **EDI standards**. The EDI Standards panel (SC01) is displayed.
2. Type **t** in the action column next to the standard you want to add the transaction set to, and press Enter.

Customizing EDI Standards

Add Copy Delete List Print Update View Elements cOmposites Segments Transactions				
SC01		EDI Standards		1 to 14 of 29
A	Standard ID	Description	Vers	Rel
-	AARV3R4	Rail Carrier Industry Version/Release 003040RAIL	03	40
-	CDTEST	UN/EDIFACT Standard (1995 Release 2) Draft 95B	95	02
-	DIUSTD	DIUSERTOADF	01	01
-	EDI891	UN/EDIFACT Standard (1989 Release 1)	89	01
t	MYEDI902	UN/EDIFACT Standard (1990 Release 2)	90	02
-	EDI94B	UN/EDIFACT Standard (1994 Release 2) Draft 94B	94	02
-	EDI95B	UN/EDIFACT Standard (1995 Release 2) Draft 95B	95	02
-	EXPANDE	Expanded DE sequence number test	01	01
-	E901EI00	UN/EDIFACT (90/1) for Electric Industry Partners	01	01
-	FAORECV	AFTSUFAO	01	01
-	ODEV3	ODETTE Standard (Version 3)	03	00
-	SAMPLE	X12 Sample for the Tutorial		
-	VICSTEST	VICS EDI STANDARDS (Version 3, Release 3)	03	30
-	VICSTST2	VICS EDI STANDARDS (Version 3, Release 3)	03	30
Command ==>				
Enter Tso F1=Help F3=Exit F8=Fwd F9=Retrieve				
F12=Cancel F13=Keys help				

The Transactions panel (SC17) is displayed.

3. Type **a** in the action column next to any transaction set, and press Enter.

Add Copy Delete List Print Update View Where used				
SC17		Transactions		1 to 12 of 19
Standard ID : MYEDI902				
Standard description : UN/EDIFACT Standard (1990 Release 2)				
A	Trans ID	Description	Group ID	
a	CONTRL	Acknowledgment/Rejection Advice Message	CONTRL	
-	CREADV	Credit Advice Message	CREADV	
-	CREEXT	Extended credit advice	CREEXT	
-	CUSDEC	Customs Declaration Message	CUSDEC	
-	CUSRES	Customs Response Message	CUSRES	
-	DEBADV	Debit Advice Message	DEBADV	
-	IFTMAN	Arrival Notice	IFTMAN	
-	IFTMBC	Booking Confirmation	IFTMBC	
-	IFTMBF	Firm Booking	IFTMBF	
-	IFTMBP	Provisional Booking	IFTMBP	
-	IFTMCS	Instruction Contract Status	IFTMCS	
-	IFTMFR	International FWDing and Transport Msg Framework	IFTMFR	

The Add Transaction panel (SC18) is displayed. The **Standard ID** and **Standard description** fields indicate which standard you are adding the transaction set to.

For EDIFACT, ODETTE, or TRADACOMS standards, a *transaction set* is the same as a message.

Block Copy Delete Get descriptions Insert Move Segments

SC18 Add Transaction 1 to 7 of 20

Standard ID : MYEDI902

Standard description : UN/EDIFACT Standard (1990 Release 2)

Transaction ID

Transaction description

Functional group ID . . .

A	Seg ID	Req Des	Max Use	Loop ID	Loop Repeat	Description
-		-				
-		-				
-		-				
-		-				
-		-				
-		-				
-		-				

4. Complete the fields as follows:

In this field:	Enter:
Transaction ID	The name of the new transaction set.
Transaction description	A brief description of the transaction set.
Functional group ID	The identifier of the functional group for this transaction set, such as IN for invoice.
Seg ID	The ID of a segment defined in the EDI standard.
Req Des	The code for the required use of the segment. Use M for mandatory, C for conditional, O for optional, or F for floating.
Max Use	The maximum number of times the segment can be used in succession at this place in the transaction. Use 9999 if the maximum use is unknown or unlimited.
Loop ID	The ID of the group of related segments, if the segment is part of a loop. See “Determining Loop IDs” on page 6-18 for more information. For EDIFACT, ODETTE, or TRADACOMS standards, a <i>loop</i> is the same as a segment group.
Loop repeat	The maximum number of times the loop that starts with this segment can be used in succession. If the loop repeat is unlimited, use 999999 . Specify this value only for the first segment of the loop.

The actions for this panel are described in Table 6-2 on page 6-7.

5. After entering the appropriate information, press Enter to save the new transaction set. The Transactions panel (SC17) is redisplayed with the new transaction set included in the list.

If you fill all the available lines and press Enter, another panel of lines is displayed. If you do not need the extra lines, go to the next step.

6. To add another transaction set, repeat steps 3 through 5.
7. Press F3 (Exit) twice to return to the Administrator's Menu (MP01).

Determining Loop IDs

There are 2 types of loop IDs used by DataInterchange: Type 1 and Type 2. Most transactions use Type 1 loop IDs.

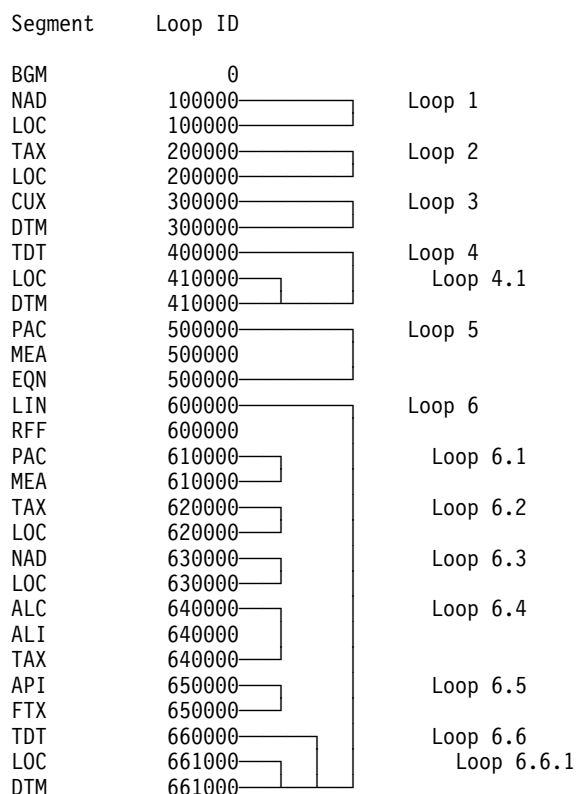
Transactions created from Application Data Formats with the **cReate** standard command, and those transactions or messages that require more than 6 nested levels of looping, require Type 2 loop IDs.

Type 1 Loop IDs:

The loop ID of a transaction segment has special significance to the mapping and translation of the transaction. DataInterchange uses the loop ID to determine where loops begin and end, and if they are nested. Loop IDs follow these rules:

- Any segment not in a loop has a loop ID of 0.
- All segments in a loop have a loop ID other than 0.
- The first loop usually starts with the loop ID of 100000, and any other segments in this loop have the same loop ID.
- To nest a loop inside loop 100000, you could use loop ID 110000.
- To create a loop at the same level as loop 100000, you could use loop ID 200000. Since the loop ID field is a hexadecimal number, the maximum number of unique loop IDs on one level is 15. If you need more than 15 at the same level, you can repeat the numbers after you use F. For example, you can use 1, 2, 3, ..., E, F, 1, 2, 3.
- To nest a loop inside loop 110000, you could use loop ID 111000. You can continue nesting in this manner until all 6 positions of the loop ID have been used.
- To create another loop nested inside loop 100000 at the same level as loop 110000, you could use loop ID 120000. Loop IDs 130000, 140000, etc. (1 thru F) would all be nested inside loop 100000 at the same level as 110000.

The following figure shows an example of looping using a modified EDIFACT invoice.



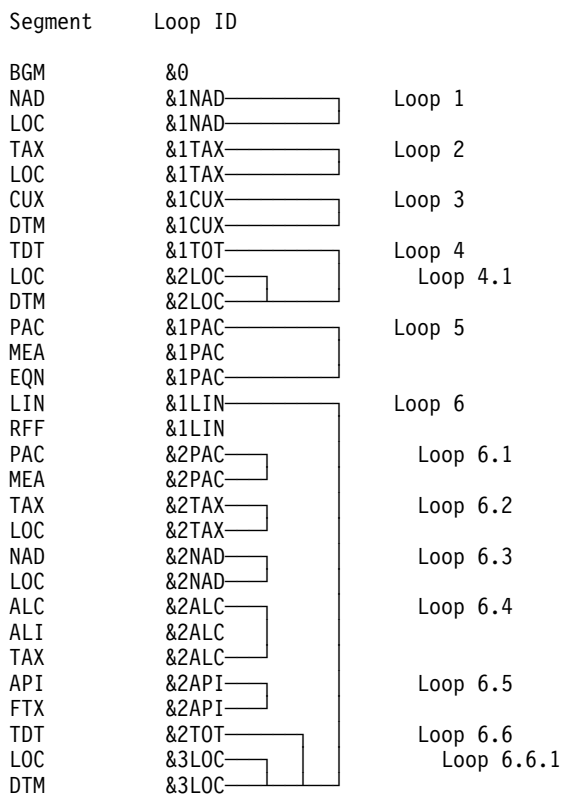
Type 2 Loop IDs:

The first position of the Type 2 loop ID is an ampersand (&), the second position is the loop level in hex (0-F), the 3rd through 6th positions are used for any 4 characters. The same loop ID must NOT be used for 2 consecutive loops. This means if you use &1ABCD for your first loop, you must not use this loop ID in another loop immediately before or after this loop.

This loop ID format is used in fixed-to-fixed mapping, allowing nested structures up to 16 levels deep (15 levels of looping plus 1 level of repeating structures).

Customizing EDI Standards

The following figure shows an example of Type 2 looping using a modified EDIFACT invoice.



Data Types for EDI Standard Data Elements

Data types define the type of data in the elements, such as alphabetic, alphanumeric, time, date, numeric, and so on. Table 6-3 on page 6-21 describes the data types you can use when customizing data elements in EDI standards.

The data types allowed for fixed standards (which are standards defined with an envelope type of F.) are allowed to have the same data types as an application data format. The application data format data types are defined in Table 7-2 on page 7-17.

Table 6-3. Data Types for EDI and Envelope Standards

Data Type	Name	Description
A	Alphabetic	All characters in the ALPHANUM table except the digits 0-9. For more information, see “ALPHANUM and CHARSET Tables” on page 8-2.
AN	Alphanumeric	All characters in the ALPHANUM table. For more information, see “ALPHANUM and CHARSET Tables” on page 8-2.
CD	Composite element	<p>The start of a subelement group. A data element with data type CD and data element ID beginning with a C for standard composite elements and S for envelope composite elements. The subelements (component elements) are defined using the <code>cOmposi tes</code> action from the standards list panel.</p> <p>For very old standards, each subelement (component element) contains an S in the Rel Def field and the reference number of the composite element or the first subelement in the related element reference numbers field.</p> <p>You can map application data to the subelements but not the type CD data element that marks the start of the composite element. If you select a subelement during mapping, the program selects the composite element automatically.</p>
DT	Date	Date format <i>yyymmdd</i> , where <i>yy</i> is the last two digits of the year, <i>mm</i> is the month (01-12), and <i>dd</i> is the day.
ID	Identifier	A data element which usually has a validation table for the valid values for the data element. For example, data element UM, unit of measure, has data type ID, and the valid values for this data element are listed in the UMCODES validation table. The table name is the same as or starts with the data element ID.
IV	Incrementing value	A data element, such as a message reference number, that starts at 1 and increases by 1 for each usage.
Nn	Numeric	<p>Numeric data with <i>n</i> places to the right of an implied decimal point. The only acceptable characters are the digits 0 through 9. N is the same as N0. For example, if the data type is N2, the value 123 means 1.23. A sign (+ or -) is optional. Positive is assumed if no sign is present. The length does not include the sign.</p> <p>The R data type should be used when defining data elements defined as data type N in EDIFACT standards.</p>
R	Real	<p>Numeric data that requires a decimal point for fractional values. The decimal point is optional for integers. A sign (+ or -) is optional. Positive is assumed if no sign is present. The length does not include the decimal point and sign.</p> <p>This data type should be used when defining data elements defined as data type N in EDIFACT standards.</p> <p><i>Examples:</i> 23.949, +23.949, -23949, -2394.9</p>
Rn	Real	Signed or unsigned numeric data with a minimum of <i>n</i> significant decimal places. On sending, at least <i>n</i> decimal places are generated. On receiving, the decimal places are the same as data type R. A sign (+ or -) is optional. Positive is assumed if no sign is present. The length does not include the decimal point and sign.
TM	Time	Time format is <i>hhmm</i> , <i>hhmmss</i> , or <i>hhmmssstt</i> , depending on the length of the data element, where <i>hh</i> is the hour, <i>mm</i> is the minutes, <i>ss</i> is the seconds, and <i>tt</i> is tenths. The time format uses a 24-hour clock, where the hour is specified as 01 to 24 for EDIFACT and 00 to 23 for X12.

Adding an EDI Standard

You may never need to add a standard. Before you consider adding a standard, determine if a currently available standard meets most of your needs. If so, copy and customize that standard instead of creating a new one.

An envelope profile has one member for each EDI standard that uses that type of enveloping. For example, if you create an EDI standard called MYEDI902 that uses type E enveloping, the mapping that uses this standard will use the MYEDI902 member of the E envelope profile as the default member.

To add an EDI standard to the database, follow these steps:

- 1. From the Administrator's Menu (MP01), select **EDI standards**. The EDI Standards panel (SC01) is displayed.
- 2. Type **a** in the action column next to any standard, and press Enter.

Add Copy Delete List Print Update View Elements cOmposites Segments Transactions

SC01

EDI Standards

1 to 14 of 29

A	Standard ID	Description	Vers	Rel
a	AARV3R4	Rail Carrier Industry Version/Release 003040RAIL	03	40
-	CDTEST	UN/EDIFACT Standard (1995 Release 2) Draft 95B	95	02
-	DIUSTD	DIUSERTOADF	01	01
-	EDI891	UN/EDIFACT Standard (1989 Release 1)	89	01
-	MYEDI902	UN/EDIFACT Standard (1990 Release 2)	90	02
-	EDI94B	UN/EDIFACT Standard (1994 Release 2) Draft 94B	94	02
-	EDI95B	UN/EDIFACT Standard (1995 Release 2) Draft 95B	95	02
-	EXPANDDE	Expanded DE sequence number test	01	01
-	E901EI00	UN/EDIFACT (90/1) for Electric Industry Partners	01	01
-	FAORECV	AFTSUFAO	01	01
-	ODEV3	ODETTE Standard (Version 3)	03	00
-	SAMPLE	X12 Sample for the Tutorial		
-	VICSTEST	VICS EDI STANDARDS (Version 3, Release 3)	03	30
-	VICSTST2	VICS EDI STANDARDS (Version 3, Release 3)	03	30

Command ==>
Enter Tso F1=Help F3=Exit F8=Fwd F9=Retrieve
F12=Cancel F13=Keys help

The Add EDI Standard panel (SC02) is displayed.

SC02

Add EDI Standard

Standard ID

Description

Version

Release

Envelope type . . .

3. Complete the fields as follows:

In this field:	Enter:
Standard ID	The name of the new standard.
Description	A brief description of the standard. This field is optional. If this is a FIXED standard (envelope type of F) and there is an application data format that is the basis for the standard, then the standard description should be the name of the application data format. Note: If the name is not the same, then Fixed-to-Fixed may not give the desired results, or generation of a Fixed-to-Fixed mapping will generate an error message.
Version	The version of the standard. This field is optional.
Release	The release of the standard. This field is optional.
Envelope type	The envelope standard used for transactions in this EDI standard (E, I, T, U, X). An envelope type value of F indicates that the standard being defined is a FIXED standard. Fixed standards have fixed length segments without delimiters between data elements. Each data element is always created with its maximum length with no delimiters used to separate the data elements or to terminate the segment.

4. After entering the appropriate information, press Enter to save the new standard. The Add Data Element panel (SC07) is displayed.
5. Complete the fields with the appropriate information, then press Enter to save the data element. The Add Data Element panel (SC07) is redisplayed. Repeat the previous step for each data element in the new standard.
6. After defining all the data elements, press F3 (Exit) twice to return to the Administrator's Menu (MP01). You can now add the segments and transaction sets to the standard using "Adding a Segment" on page 6-13 and "Adding a Transaction Set" on page 6-15.

Viewing Standards, Transaction Sets, Segments, Composite Elements, and Data Elements

To view standards, transaction sets, segments, composite elements, and data elements, follow these steps:

From the Administrator's Menu (MP01), select **EDI standards** to display the EDI Standards panel (SC01).

To view a **standard**:

1. Type **v** in the action column next to the standard you want to view, and press Enter. The View EDI Standard panel (SC05) is displayed.

Customizing EDI Standards

SC05	View EDI Standard
Standard ID	MYEDI902
Description	UN/EDIFACT Standard (1990 Release 2)
Version	01
Release	00
Envelope type	E

2. Press F3 (Exit) to return to the EDI Standards panel (SC01).

To view a **transaction set**:

1. Type **t** in the action column next to the standard containing the transaction set you want to view, and press Enter. The Transactions panel (SC17) is displayed.
2. Type **v** in the action column next to the transaction set you want to view, and press Enter. The View Transaction panel (SC21) is displayed.

SC21	View Transaction				1 to 9 of 26
Standard ID : MYEDI902					
Standard description : UN/EDIFACT Standard (1990 Release 2)					
Transaction ID : CREADV					
Transaction description : Credit Advice Message					
Functional group ID . . : CREADV					
Seg ID	Req Des	Max Use	Loop ID	Loop Repeat	Description
BGM	M	1	0	0	Beginning of Message
BUS	C	1	0	0	Business Function
NAD	M	1	001000	6	Name and Address
CTA	C	1	001000	0	Contact Information
COM	C	1	001000	0	Communication Contacts
FII	M	1	002000	4	Financial Institution Information
CTA	C	1	002000	0	Contact Information
COM	C	1	002000	0	Communication Contacts
DTM	M	3	0	0	Date/Time/Period

3. Press F3 (Exit) twice to return to the EDI Standards panel (SC01).

To view a **segment**:

1. Type **s** in the action column next to the standard containing the segment you want to view, and press Enter. The Segments Directory panel (SC11) is displayed.
2. Type **v** in the action column next to the segment you want to view, and press Enter. The View Segment panel (SC15) is displayed.

SC15	View Segment	1 to 2 of 2			
Standard ID : MYEDI902					
Standard description : UN/EDIFACT Standard (1990 Release 2)					
Segment ID : ACA					
Segment description . : Alternative Currency Amount _____					
Ref	Element	Req	Rel	Related Element	Description
Num	ID	Des	Def	Reference Numbers	
1	C275	C			Alternative currency
2	5004	M			Monetary amount

3. Press F3 (Exit) twice to return to the EDI Standards panel (SC01).

To view a **composite element**:

1. Type **o** in the action column next to the standard containing the composite element you want to view, and press Enter. The Composite Element Dictionary panel (SC30) is displayed.
2. Type **v** in the action column next to the composite element you want to view, and press Enter. The View Composite Element panel (SC34) is displayed.

SC34	View Composite Element	1 to 4 of 4		
Standard ID : MYEDI902				
Standard description : UN/EDIFACT Standard (1990 Release 2)				
Composite ID : C002				
Composite description : Document/message name				
Ref	Element	Req Rel Related Element Description		
Num	ID	Des Def Reference Numbers		
1	1001	C		Document/message name, code
2	1131	C		Code list qualifier
3	3055	C		Code list responsible agency
4	1000	C		Document/message name

3. Press F3 (Exit) twice to return to the EDI Standards panel (SC01).

To view a **data element**:

1. Type **e** in the action column next to the standard containing the data element you want to view, and press Enter. The Data Element Dictionary panel (SC06) is displayed.
2. Type **v** in the action column next to the data element you want to view, and press Enter. The View Data Element panel (SC10) is displayed.

SC10	View Data Element
Standard ID	MYEDI902
Standard description	UN/EDIFACT Standard (1990 Release 2)
Data element ID	C003
Description	Additional Entry Details
Data type	CD
Minimum length	1
Maximum length	1
Validation table	

3. Press F3 (Exit) twice to return to the EDI Standards panel (SC01).
4. Press F3 (Exit) to return to the Administrator's Menu (MP01).

Customizing Envelope Standards

DataInterchange also lets you customize the envelope standards, which consist of the segments and data elements making up the headers and trailers that enclose transaction sets, functional groups, and interchanges. You can also define the delimiters used in the standard, such as the character that separates data elements. When you select **Envelope standards** from the Administrator's Menu (MP01), the system displays the Envelope Standards panel (SC16), which lists the envelope standards in your DataInterchange database. If the Add Envelope Standard panel (SC26) is displayed, or if the envelope you are looking for is not listed on the Envelope Standards panel (SC16), you can use the Import facility to add it. For more information, see "Importing EDI and Envelope Standards" on page 3-1. You can work with an entire standard, or use the commands on the action bar to work with the segments or data elements in the standard.

Providing Envelope Data

Envelope standards define the data elements and segments that are used for enveloping your transactions. The data to build the envelopes comes from the sources shown in Figure 6-3.

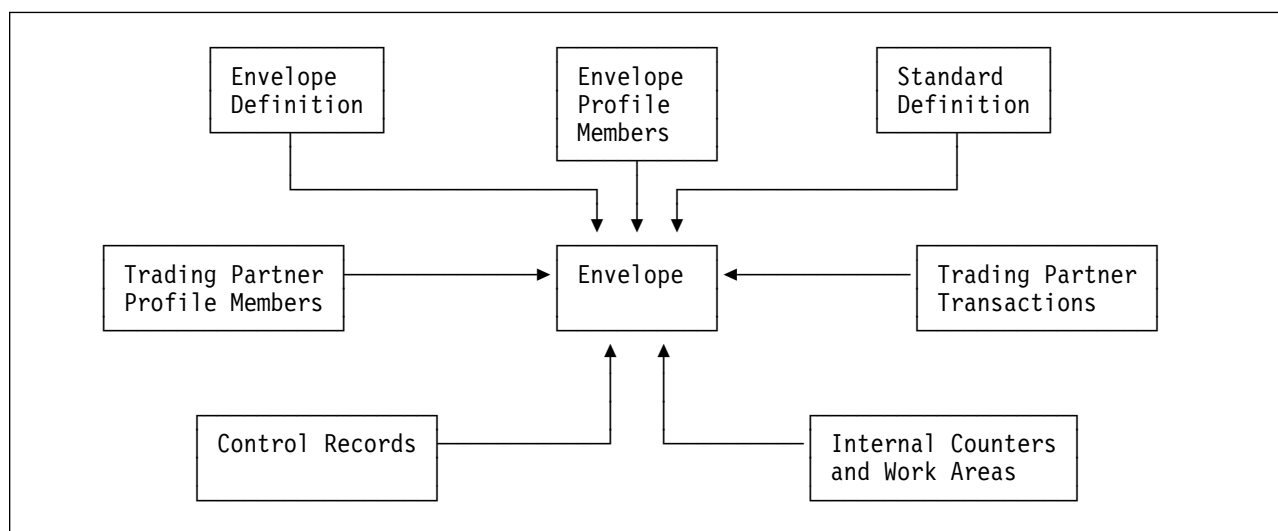


Figure 6-3. Data Sources for Envelopes

Data Source	What It Does	Where to Find More Information
Envelope Profile Member	Supplies data that is the same every time the member is used; for example, the sender ID, the syntax ID, and the syntax qualifier	"Envelope Profiles" on page 6-30
Envelope Definition	Provides delimiters and enveloping segment IDs	"Updating an Envelope Standard" on page 6-27
Standard Transaction Definition	Provides the functional group ID	"Adding a Transaction Set" on page 6-15
Trading Partner Profile Member	Supplies data that is the same every time you send to a specific trading partner; for example, the interchange ID and password	"Setting Up the Trading Partner Profile" on page 5-23

Data Source	What It Does	Where to Find More Information
Trading Partner Transaction	Provides data that is specific to the transactions in the envelope; for example, the EDI standard transaction set ID, envelope type, and overrides	"What is a Trading Partner Transaction?" on page 9-1
Internal Counter and Work Area	Supplies data that changes with each enveloping procedure; for example, the date and time, or total transactions in the envelope	Table 6-4 on page 6-33
Control Record	Supplies data such as the interchange ID and password, application sender and receiver IDs, application reference, and the version and release of an EDI standard. Data provided by a control record passed to the Transaction Store Utility overrides data from any other source.	<i>DataInterchange Programmer's Reference</i>

Updating an Envelope Standard

Update an envelope standard if you need to change the element separator, subelement separator, segment terminator, decimal notation, release character, or segment ID separator. You can also change the segment IDs for envelope headers and trailers.

To update an envelope standard, follow these steps:

1. From the Administrator's Menu (MP01), select **Envelope standards**. The Envelope Standards panel (SC16) is displayed.

Add Copy Delete Print Update View Elements cOmposites Segments Generate		
SC16		Envelope Standards 1 to 5 of 5
A	Envelope Type	Description
<input checked="" type="checkbox"/>	E	EDIFACT delimiter and envelope definitions
<input type="checkbox"/>	I	ICS delimiter and envelope definitions
<input type="checkbox"/>	T	UNTDI delimiter and envelope definitions
<input type="checkbox"/>	U	UCS delimiter and envelope definitions
<input type="checkbox"/>	X	X12 delimiter and envelope definitions

Customizing EDI and Envelope Standards

2. Type **u** in the action column next to the envelope standard you want to update, and press Enter. The Update Envelope Standard panel (SC28) is displayed. The highlighted fields on your screen are the fields that you can change.

SC28	Update Envelope Standard				
Envelope type :	E				
Description	EDIFACT delimiter and envelope definitions_____				
Group optional (Y/N)	Y				
Delimiter segment ID	UNA_____				
Delimiters		Hex	Character	Hex	Character
Element separator . .	—	+	*Decimal notation . . .	—	.
Sub-elem separator . .	—	:	*Release character . . .	—	?
Segment terminator . .	—	'	*Segment ID separator	—	_
Envelopes		Segment ID	Segment ID		
Interchange header . .	:	UNB	Interchange trailer :	UNZ	
Group header	:	UNG	Group trailer	UNE	
Transaction header . .	:	UNH	Transaction trailer	UNT	
*Not used for all envelope types					

The first two lines identify the envelope standard you selected. *Group optional* applies only to envelope types that do not require functional group headers and trailers. **Delimiter segment ID** indicates the segment used to send your delimiter values to a trading partner. In the example above, the UNA segment in the EDIFACT standard is used.

The *delimiters* section defines some special characters for transaction sets that use this envelope definition. You can customize this portion of the standard by typing the delimiter characters or their EBCDIC hexadecimal values.

The bottom section shows the IDs of the segments this envelope definition uses. You can change the segment IDs of the group and transaction set headers and trailers, but not the segment IDs of the interchange header and trailer. Ordinarily, you do not change any of them. If you do, the segments you identify must exist in this envelope standard.

3. Place your cursor on an entry field and press F1 (Help) to view the explanation of that field. A colon precedes fields you cannot change, and help is not available for these fields.
4. Type the new information in the fields you want to update, and press Enter to save your changes.
5. Or press F12 (Cancel) to leave the panel without saving your changes. The Envelope Standards panel (SC16) is redisplayed.
6. Press F3 (Exit) to return to the Administrator's Menu (MP01).

Updating Envelope Segments

To update an envelope segment, follow these steps:

1. From the Administrator's Menu (MP01), select **Envelope standards**. The Envelope Standards panel (SC16) is displayed.
2. Type **s** in the action column next to the standard you want to update, and press Enter. The Segment Directory panel (SC11) is displayed.

Add Copy Delete Print Update View Elements cOmposites Segments Generate		
SC16		Envelope Standards 1 to 5 of 5
A	Envelope Type	Description
s	E	EDIFACT delimiter and envelope definitions
-	I	ICS delimiter and envelope definitions
-	T	UNTDI delimiter and envelope definitions
-	U	UCS delimiter and envelope definitions
-	X	X12 delimiter and envelope definitions

3. Type **u** in the action column next to the segment you want to update, and press Enter. The Update Segment panel (SC14) is displayed.

Block Copy Delete Get descriptions Insert Move Elements						
SC14			Update Segment			1 to 9 of 11
Standard ID : E						
Standard description : EDIFACT delimiter and envelope definitions						
Segment ID. : UNB						
Segment description . . Interchange header_____						
A	Ref Num	Element ID	Req Des	Rel Def	Related Element Reference Numbers	Description
-	1	S001	M	-	_____	Syntax Identifier
-	2	S002	M	-	_____	Interchange Sender
-	3	S003	M	-	_____	Interchange Recipient
-	4	S004	M	-	_____	Date and Time of Preparation
-	5	0020	M	-	_____	Interchange Control Reference
-	6	S005	C	-	_____	Recipient Reference/Password
-	7	0026	C	-	_____	Application Reference
-	8	0029	C	-	_____	Processing Priority Code
-	9	0031	C	-	_____	Acknowledgement Request
Command ==>						
Enter Tso F1=Help F3=Exit F8=Fwd F9=Retrieve						
F12=Cancel F13=Keys help						

4. Make the changes, and press Enter.
5. Press F3 (Exit) twice to return to the Administrator's Menu (MP01).

Using Data Types in Envelope Standards

DataInterchange defines special data types that appear only in envelope standards to indicate how a data element is being used and the source of the envelope data.

To update the data type of an envelope data element, follow these steps:

1. From the Administrator's Menu (MP01), select **Envelope standards**.
2. On the Envelope Standards panel (SC16), type **e** in the action column next to the standard you want to update, and press Enter. The Data Element Dictionary panel (SC06) is displayed.

Customizing Envelope Standards

3. Type **u** in the action column next to the data element you want to update, and press Enter. The Update Data Element panel (SC09) is displayed.

SC09	Update Data Element
Standard ID	E
Standard description	EDIFACT delimiter and envelope definitions
Data element ID	0002
Description	Syntax Version Number _____
Data type	N0
Minimum length	1
Maximum length	1
Validation table	_____

4. Change the data type, and press Enter. Data types for envelopes are described in Table 6-4 on page 6-33.
5. Press F3 (Exit) twice to return to the Administrator's Menu (MP01).

Envelope Profiles

DataInterchange provides one envelope profile for each envelope standard, and the profile IDs match the envelope standard IDs: E, I, T, U, and X. An envelope profile has one member for each EDI standard that uses that type of enveloping. For example, if you create an EDI standard called MYEDI902 that uses type E enveloping, the E envelope profile uses MYEDI902 as the *default* member. You can also add members with names that do not match an EDI standard ID, and then use these names when mapping a trading partner transaction to override the default member. If you do not provide a default member, you must provide an override for each send usage using this envelope.

The envelope profiles have one field for each data element in the envelope standard. The profile members provide literal or constant data for building header and trailer segments for transaction sets, messages, functional groups, and interchanges. Therefore, you must supply only the values that need to be populated and for which a value is not provided by another source. See the following sections for the envelope standard you are using. You will need to customize the sender ID field of the envelope profile member you will be using. If you use more than one sender ID, you will need to create an additional envelope profile member for each sender ID you use.

The first field in the profile is the name of the member. The remaining fields represent the data elements in the envelope standard. The field names are designed to make cross-referencing easy. For example, field UNB03 is the third data element in the UNB segment.

- | A generic envelope profile member name can consist of 1 to 6 characters (base name). When a generic envelope profile is accessed by the trading partner usages, DataInterchange appends the envelope profile suffix from the trading partner profile to the base name to determine which profile to access during enveloping.

Adding a Member to an Envelope Profile

To add a member to an envelope profile, follow these steps:

1. From the Administrator's Menu (MP01), select **Profiles**. The Profile Definitions panel (PM01) is displayed.

2. Type **m** in the action column next to the profile you want to add a member to, and press Enter. The Profile Members panel (PM07) is displayed.

Log work with Members Print View			
PM01		Profile Definitions	1 to 17 of 17
Action	Profile ID	Profile Description	Log?
-	ACTLOGS	Activity log	N
-	ADAMCTL	User program information	N
-	APPDEFS	Application Definition Profile	N
-	CONTRECV	Continuous Receive Profile	N
m	E	EDIFACT standard envelope data	N
-	I	ICS standard envelope data	N
-	LANGPROF	Language profile	N
-	MQSERIES	MQSeries Queue profile	N
-	NETOP	Network operation profile	N
-	NETPROF	Network profile	N
-	REQPROF	Requestor profile	N
-	SECUPROF	Security profile	N
-	SYSPROF	System profile	N
-	T	UN/TDI standard envelope data	N
-	TPPROF	Trading partner profile	Y
-	U	UCS standard envelope data	N
-	X	X12 standard envelope data	N

3. Type **a** in the action column next to any member, and press Enter.

Add Copy Delete List Print Update View			
PM01		Profile Definitions	1 to 17 of 17
Action	Profile ID	PM07 Profile Members 1 to 9 of 13	
-	ACTLOGS	Profile ID : E	
-	ADAMCTL	Description: EDIFACT standard envelope data	
-	APPDEFS		
-	CONTRECV		
m	E	Action Key	
-	I	a	EDIV1R0
-	LANGPROF	-	EDI891
-	MQSERIES	-	EDI892
-	NETOP	-	EDI901
-	NETPROF	-	MYEDI90S
-	REQPROF	-	EDI902
-	SECUPROF	-	EDI911
-	SYSPROF	-	EDI912
-	T		
-	TPPROF		
-	U	UCS standard envelope data	N
-	X	X12 standard envelope data	N

The Add Profile Member panel (PM08) is displayed.

Customizing Envelope Standards

PM08	Add Profile Member	1 to 13 of 47
Profile ID: E		
Fill in the information below and press Enter to save this member. To stop entering members, press Exit or Cancel.		
EDIFACTKEY	_____
UNB01	_____
UNB02	_____
UNB03	_____
UNB04	_____
UNB05	_____
UNB06	_____
UNB07	_____
UNB08	_____
UNB09	_____
UNB10	_____
UNB11	_____
UNB12	_____

4. Type the appropriate information, if any, in each field. If you need help with a field, place the cursor in the field, then press F1 (Help).
5. Press F3 (Exit) when you are finished viewing the help information. The envelope profile data is also described in “Envelope Profiles” on page 6-30. Use Forward (F8) to go the next panel of profile fields.
6. Press Enter to save the new member after completing the necessary fields. The Profile Members panel (PM07) is redisplayed with the new member included in the list.
7. Press F3 (Exit) twice to return to the Administrator’s Menu (MP01).

Data Types for Envelope Standards

The data types in the following table define the type of data in a data element, tell the enveloper where to look for the data for that element during send processing, and tell the deenveloper what to do with the data during receive processing.

Table 6-4 (Page 1 of 3). Data Types for Envelope Standards Only

Data Type	Name	Description
AG	Responsible agency code	<p>Identifies the field within the service segments that provides the code for the agency responsible for the transaction or message definition. This is used during inbound processing to determine a receive usage (map) that should be used to translate the transaction. If agency codes will be used, then it is suggested that the Envelope Standards option be chosen and that data element 455 in the X, U, and I standards and data element 0051 in the E standard be changed to have an AG data type.</p> <p>It is also suggested that the agency code be combined with the version and release so that different routing (or a completely different map) would be possible based on the defining agency and the version of the standard transaction.</p> <ul style="list-style-type: none"> Version - This field may be used in addition to the application sender and application receiver IDs to provide additional routing capabilities. It is expected, but not necessary, that version would be combined with the responsible agency code and release fields, so that different routing (or a completely different map) would be possible based on the defining agency and the version of the standard transaction. Release - This field may be used in addition to the application sender and application receiver IDs to provide additional routing capabilities. It is expected, but not necessary, that release would be combined with the responsible agency code and release so that different routing (or a completely different map) would be possible based on the defining agency, version, and release of a transaction. For example, it would be possible to have a separate map for an X1 Version 3 Release 1 810 transaction and an X12 Version 3 Release 2 81 transaction from the same trading partner. The data entered into this field must match exactly the release when taken from the received transaction, except for trailing blanks, which are not significant.
AP	Application reference	The name of the recipient's application area that is to receive the interchange messages.
AR	Application receiver	The name of the receiver's application area that is to receive the group of transactions. This can be part of the key to the trading partner transaction receive usage.
AS	Application sender	The name of the sender's application area. This can be part of the key to the trading partner transaction receive usage. For sending, the enveloper places the application data format ID in the field with the AS data type. To disable this function for sending, change the data type to AN.
CN	Control number	A number used to reference the envelope. It can also be used to determine an envelope if control numbers are used sequentially. The control number in a header segment has a matching control number in the corresponding trailer.
CO	Company Name	The recipient's company name. When an envelope is built, the value from the 'Company Name' field of the trading partner profile will be used as source for the field with this data type. If no value is supplied in the trading partner profile, then the value from the envelope profile (X, E, I, T, U or F) field will be used. No envelope as shipped by DataInterchange has a field with this data type.

Customizing Envelope Standards

Table 6-4 (Page 2 of 3). Data Types for Envelope Standards Only

Data Type	Name	Description
CT	Count	<p>The number of entities at the next lower level in the envelope. For example:</p> <ul style="list-style-type: none"> • The number of segments in a transaction • The number of transactions in a group • The number of groups in an interchange <p>Note: For EDIFACT enveloping, the UNZ01 field (0036) is defined to be the number of groups, if groups are used in the interchange. Otherwise, it is the number of messages. The CT data type handles this situation, because if groups are not used, the count of messages becomes the next lower level to the interchange.</p>
FG	Functional group	A functional group ID, such as IN for invoices. For enveloping, the functional group ID from the standard transaction is used.
IR	Interchange receiver	The identification of the trading partner who is receiving the interchange envelope.
IS	Interchange sender	The identification of the trading partner who is sending the interchange envelope.
IV	Incrementing value	A data element, such as a message reference number, that starts at 1 and increments by 1 for each usage. Can be used for explicit nesting in EDIFACT and UNTDI interchanges.
LV	Release level	The release level of the standard that contains this data element.
RS	Routing sender	<p>The sender's application area. This is similar to the AS data type. This can be part of the key to the trading partner transaction receive usage. If an AS field is received, it overrides the RS field. For more information, see the Application routing (sending) field description on page 9-90.</p> <p>This data type is commonly used to define one of the data elements in the interchange header as RS. When a trading partner sends you envelopes without functional groups, you can use application information from the interchange envelope to route the transaction or message.</p>
RR	Routing receiver	<p>The receiver's application area. This is similar to the AR data type. This can be part of the key to the trading partner transaction receive usage. If an AR field is received, it overrides the RR field. For more information, see the Application routing (receiver) field description on page 9-90.</p> <p>This data type is commonly used to define one of the data elements in the interchange header as RR. When a trading partner sends you envelopes without functional groups, you can use application information from the interchange envelope to route the transaction or message.</p>
PW	Password	A password used in the interchange or functional group header.
TC	Transaction code	A transaction identifier specified by the standard, such as 850 for an X12 purchase order.
TI	Test indicator	An indication of whether or not the transaction data is for testing only.
TS	Total segments	The number of segments in the interchange, including the interchange header and the interchange trailer.

Table 6-4 (Page 3 of 3). Data Types for Envelope Standards Only

Data Type	Name	Description
TT	Total transactions	<p>The number of transaction (message) sets between the interchange header and interchange trailer.</p> <p>Note: For UNTDI enveloping, the END01 field (016) is defined to be the total number of messages in the interchange, which is exactly what the TT data type provides. However, DataInterchange will also allow the END01 field to have a datatype of CT, because groups are rarely used in UNTDI, and without groups, the CT and TT data types provide the same function.</p>
VR	Version	The version of the standard that contains this data element.

If you use one of the following data types for an envelope data element, the enveloper looks for the data in the envelope profile:

- Alphabetic (A)
- Alphanumeric (AN)
- Composite element (CD)
- Identifier (ID)
- Numeric (Nn)
- Real (Rn)

These data elements are described in Table 6-3 on page 6-21.

EDIFACT Envelope Data

The E envelope profile contains data for EDIFACT interchange envelopes. In the Data Source column:

P = Profile member is the primary source.

S = Supplied with product.

L = Logic used to fill field. See Description column.

Table 6-5 (Page 1 of 7). EDIFACT Envelope Data

Field Label	Data Source	Element ID	Length	Data Type	Description
EDIFACTKEY			008		<p>Profile retrieval key</p> <p>The name of this member. For each standard you define with type E enveloping, you must provide a member whose name matches the standard ID. You can provide additional members with names that do not match a standard ID and use these names as overrides when mapping a trading partner transaction. The transaction mapping field Standard profile member name specifies the name of the override member.</p>
Interchange Header Segment (UNB)					
UNB01	PS	0001	004	A	<p>Syntax identifier</p> <p>The identification of the agency controlling the syntax being used. The controlling agency is UNO. The level is A or B. (IBM Global Network users specify A. The field would therefore contain UNOA.) The LV or VR data type in the envelope definition signals the enveloper to use the IVERREL field of the control record as the syntax identifier and ignore this field.</p>

Customizing Envelope Standards

Table 6-5 (Page 2 of 7). EDIFACT Envelope Data

Field Label	Data Source	Element ID	Length	Data Type	Description
UNB02	PS	0002	001	R	<p>Syntax version number</p> <p>The version number of the syntax identified in UNB01. The LV or VR data type in the envelope definition signals the enveloper to use the IVERREL field of the control record as the syntax version and ignore this field.</p>
UNB03	PS	0004	035	IS	<p>Sender ID</p> <p>The name or code that identifies the sender. If a code is used, enter its qualifier in UNB04. The IS data type in the envelope standard signals the enveloper to use the ISID field of the control record as the sender ID. If ISID is blank, UNB03 is used by default.</p>
UNB04	PS	0007	004	AN	<p>Sender ID qualifier</p> <p>The qualifier for the sender ID code in UNB03. Refer to the data element directory for a list of code qualifiers. A qualifier of ZZ indicates that UNB03 is a network account code and user ID.</p>
UNB05	P	0008	014	AN	<p>Reverse routing address</p> <p>The address the receiver includes which routes it to the appropriate department in your company, such as a return address on an envelope. IBM Global Network does not use this field.</p>
UNB06	L	0010	035	IR	<p>Recipient ID</p> <p>The name or code that identifies the receiver. If you use a code, enter its qualifier in UNB07. The IR data type in the envelope definition signals the enveloper to use the following sequence for supplying the ID:</p> <ul style="list-style-type: none"> If the default profile member is used: <ol style="list-style-type: none"> IRID field from the control record passed to the translator, if used Interchange ID from the trading partner profile Account number/user ID from the trading partner profile Value in UNB06 If the override profile member is used: <ol style="list-style-type: none"> IRID field from the control record passed to the translator, if used Value in UNB06 Interchange ID from the trading partner profile Account number/user ID from the trading partner profile
UNB07	L	0007	004	AN	<p>Recipient ID qualifier</p> <p>The qualifier for the recipient ID code in UNB06. Refer to the data element directory for a list of code qualifiers. If the trading partner profile contains an interchange ID and interchange qualifier, the enveloper uses the interchange qualifier from the trading partner profile and ignores this field.</p>

Table 6-5 (Page 3 of 7). EDIFACT Envelope Data

Field Label	Data Source	Element ID	Length	Data Type	Description
UNB08	P	0014	014	AN	Routing address The address the recipient uses to route the transactions within the recipient's company.
UNB09	L	0017	006	DT	Date The date when the interchange was prepared. The DT data type in the envelope definition signals the enveloper to use the system date and ignore this field.
UNB10	L	0019	004	TM	Time The time at which the interchange was prepared. The TM data type in the envelope definition signals the enveloper to use the system time and ignore this field.
UNB11	L	0020	014	CN	Interchange control reference A number used to reference the envelope. It must match UNZ02. The CN data type in the envelope definition signals the enveloper to use the interchange control number from the trading partner profile and ignore this field.
UNB12	L	0022	014	PW	Password A password assigned by the recipient, to be used as agreed upon by the trading partners. The PW data type in the envelope definition signals the enveloper to use the interchange password from the trading partner profile and ignore this field. You can override the password by including one in the IPSW field of the control record passed to the translator.
UNB13	L	0025	002	AN	Password qualifier A qualifier to the recipient's password, to be used as agreed upon by the trading partners. The enveloper sets the qualifier to 01 if a password is present or to blanks if one is not. It uses this field only if the envelope definition does not contain a PW data type.
UNB14	P	0026	014	AP	Application reference The sender's identification of the functional area to which the interchange messages relate. The message identifier, for example, might be used here if the interchange contains only one type of message. The AP data type in the envelope definition signals the enveloper to use the IAPREF field of the control record as the application reference and ignore this field. The TC data type in the envelope definition signals the enveloper to use the standard transaction ID of the first message as the application reference and ignore this field. If you change the data type to AP, you can also use this field, when sending the envelope over the network, to specify the message user class, instead of using a value in the <i>message user class</i> field of the requestor profile member. If you specify a value in the requestor profile member, it overrides the value specified in this field. The value in the envelope does not change.

Customizing Envelope Standards

Table 6-5 (Page 4 of 7). EDIFACT Envelope Data

Field Label	Data Source	Element ID	Length	Data Type	Description
UNB15	P	0029	001	A	Processing priority The sender's code for processing priority, as agreed upon with the trading partner. Code A is highest priority.
UNB16	P	0031	001	AN	Acknowledgment request The sender's code for requesting an acknowledgment.
UNB17	P	0032	035	AN	Communications agreement ID The name or code for the type of agreement used for this interchange, as agreed to with the trading partner.
UNB18	L	0035	001	TI	Test indicator An indication that the interchange is for testing, where 1 indicates a test interchange. The TI data type in the envelope definition signals the enveloper to use the TESTIND field from the control record or the RAWTEST keyword (for raw data) and ignore this field.
Interchange Trailer Segment (UNZ)					
UNZ01	L	0036	006	CT	Interchange control count A control total of the number of functional groups (if used) or messages in the interchange. The CT data type in the envelope definition signals the enveloper to use the internal counter and ignore this field.
UNZ02	L	0020	014	CN	Interchange control reference A number used to reference the envelope. It must match UNB11. The CN data type in the envelope definition signals the enveloper to use the same value that was used for UNB11.
Functional Group Header Segment (UNG)					
UNG01	L	0038	006	FG	Functional group ID An identification of the type of messages in the group. The FG data type in the envelope definition signals the enveloper to use the functional group ID from the standard transaction and ignore this field.
UNG02	L	0040	035	AS	Sender ID The name or code for a specific department in the sender's company. The AS data type in the envelope definition signals the enveloper to use other sources for the sender ID. The sources, in priority sequence, are: <ol style="list-style-type: none"> 1. GSID in the control record, if used 2. Application sender ID in the trading partner transaction 3. UNG02 (this field) 4. The data format ID You can pair the sender ID with a qualifier in UNG03.
UNG03	P	0007	004	AN	Sender ID qualifier The qualifier for the sender ID code in UNG02. Refer to the data element directory for a list of code qualifiers.

Table 6-5 (Page 5 of 7). EDIFACT Envelope Data

Field Label	Data Source	Element ID	Length	Data Type	Description
UNG04	L	0044	035	AR	<p>Recipient ID</p> <p>The name or code for the specific department in the recipient's company that is to receive the group. The AR data type in the envelope definition signals the enveloper to use other sources for the recipient ID. The sources, in priority sequence, are:</p> <ol style="list-style-type: none"> 1. GSID in the control record, if used 2. Application receiver ID in the trading partner transaction 3. UNG04 (this field) <p>You can pair the recipient ID with a qualifier in UNG05.</p>
UNG05	P	0007	004	AN	<p>Recipient ID qualifier</p> <p>The qualifier for the recipient ID code in UNG04. Refer to the data element directory for a list of code qualifiers.</p>
UNG06	L	0017	006	DT	<p>Date</p> <p>The date when the functional group was prepared. The DT data type in the envelope definition signals the enveloper to use the system date and ignore this field.</p>
UNG07	L	0019	004	TM	<p>Time</p> <p>The time at which the functional group was prepared. The TM data type in the envelope definition signals the enveloper to use the system time and ignore this field.</p>
UNG08	L	0048	014	CN	<p>Functional group reference number</p> <p>A number used to reference the envelope. It must match UNE02. The CN data type in the envelope definition signals the enveloper to use the functional group control number from the trading partner profile and ignore this field. The IV data type can also be used. It signals the enveloper to start at 1 and sequentially number the groups in the interchange.</p>
UNG09	PS	0051	002	AN	<p>Controlling agency</p> <p>The code that identifies the agency that has control of the message type in the functional group.</p>
UNG10	PS	0052	003	AN	<p>Message version</p> <p>The version number for the message type. You can override the version number by providing it in the GVER field of the control record passed to the translator. It maps to the VR data type in the envelope definition.</p>
UNG11	PS	0054	003	AN	<p>Message release</p> <p>The release number within the version number for the message type. You can override the release number by providing it in the GREL field of the control record passed to the translator. It maps to the LV data type in the envelope definition.</p>
UNG12	P	0057	006	AN	<p>Association assigned</p> <p>The code, assigned by the responsible association, that further identifies the message type.</p>

Customizing Envelope Standards

Table 6-5 (Page 6 of 7). EDIFACT Envelope Data

Field Label	Data Source	Element ID	Length	Data Type	Description
UNG13	P	0058	014	AN	<p>Application password</p> <p>The password assigned by the specific department in the recipient's company. The PW data type in the envelope definition signals the enveloper to look first in the GAPW field of the control record and then in the trading partner usage overrides.</p>
Functional Group Trailer Segment (UNE)					
UNE01	L	0060	006	CT	<p>Number of messages</p> <p>A control total of the number of messages in a functional group. The CT data type in the envelope definition signals the enveloper to use its internal counter and ignore this field.</p>
UNE02	L	0048	014	CN	<p>Functional group reference number</p> <p>A number used to reference the envelope. It must match UNG08. The CN or IV data type in the envelope definition signals the enveloper to use the same value that was used for UNG08.</p>
Message Header Segment (UNH)					
UNH01	L	0062	014	CN	<p>Message reference number</p> <p>A number used to reference the envelope. It must match UNT02. The CN data type in the envelope definition signals the enveloper to use the transaction set control number from the trading partner profile and ignore this field. The IV data type can also be used. It signals the enveloper to start at 1 and sequentially number the transactions in the group or in the interchange envelope if there are no groups.</p>
UNH02	L	0065	006	TC	<p>Message type</p> <p>A code assigned by the controlling agency to identify the message type. The TC data type in the envelope definition signals the enveloper to use the standard transaction ID and ignore this field.</p>
UNH03	P	0052	003	VR	<p>Message version number</p> <p>The version number for the message type. You can override the version number by providing it in the TVER field of the control record passed to the translator. It maps to the VR data type in the envelope definition.</p>
UNH04	P	0054	003	AN	<p>Message release number</p> <p>The release number within the version number for the message type. You can override the release number by providing it in the TREL field of the control record passed to the translator. It maps to the LV data type in the envelope definition.</p> <p>Use the Envelope standards option on the main menu to change the data type of this field to LV if 'Release' is used to qualify receive usages.</p>

Table 6-5 (Page 7 of 7). EDIFACT Envelope Data

Field Label	Data Source	Element ID	Length	Data Type	Description						
UNH05	P	0051	002	AN	<p>Controlling agency</p> <p>A code for the agency that has control of the message type. Refer to the standards manual (data element 0051) for these codes.</p> <p>Use the Envelope standards option on the main menu to change the data type of this field to AG if 'Responsible agency code' is used to qualify receive usages.</p>						
UNH06	P	0057	006	AN	<p>Association assigned</p> <p>The code, assigned by the responsible association, that further identifies the message type. Refer to the standards manual (data element 0057) for these codes.</p>						
UNH07	P	0068	035	AN	<p>Common access reference</p> <p>The key that relates all subsequent transfers of data to a common file. Trading partners can agree to using a key made up of components, but subelement separators cannot be used.</p>						
UNH08	P	0070	002	R	<p>Sequence of transfer</p> <p>A number the sender assigns to indicate that the message is an addition or change to a previous message. The first message in a sequence is number 1.</p>						
UNH09	P	0073	001	A	<p>First and last</p> <p>A code for the first and last message in a sequence of the same type and relating to the same topic:</p> <table><thead><tr><th>Code</th><th>Description</th></tr></thead><tbody><tr><td>C</td><td>For creation. C must be present for the first transfer if more than one is foreseen.</td></tr><tr><td>F</td><td>For final. F must be present for the last transfer of a sequence.</td></tr></tbody></table>	Code	Description	C	For creation. C must be present for the first transfer if more than one is foreseen.	F	For final. F must be present for the last transfer of a sequence.
Code	Description										
C	For creation. C must be present for the first transfer if more than one is foreseen.										
F	For final. F must be present for the last transfer of a sequence.										
Message Trailer Segment (UNT)											
UNT01	L	0074	006	CT	<p>Number of segments in message</p> <p>A control total of the number of segments in a message, including the UNH and UNT segments. The CT data type in the envelope definition signals the enveloper to use its internal counter and ignore this field.</p>						
UNT02	L	0062	014	CN	<p>Message reference number</p> <p>A number used to reference the envelope. It must match UNH01. The CN or IV data type in the envelope definition signals the enveloper to use the same value that was used for UNH01.</p>						

ICS Envelope Data

The I envelope profile contains data for ICS interchange envelopes. In the Data Source column:

P = Profile member is the primary source.

S = Supplied with product.

L = Logic used to fill field. See Description column.

Table 6-6 (Page 1 of 5). ICS Envelope Data

Field Label	Data Source	Element ID	Length	Data Type	Description
ICSKEY			008		Profile retrieval key The name of this member. For each standard you define with type I enveloping, you must provide a member whose name matches the standard ID. You can provide additional members with names that do not match a standard ID and use these names as overrides when mapping a trading partner transaction.
Interchange Header Segment (ICS)					
ICS01	L	X02	001	AN	Sub element separator The character that separates subelements in a composite element. The translator uses the character specified in the trading partner profile or, if not specified there, it uses the character specified in the envelope standard.
ICS02	P	X03	004	ID	Control standards identifier A code for the agency that controls the interchange syntax. Refer to the description of data element X03 in the standards manual for these IDs. The LV or VR data type in the envelope definition signals the enveloper to use the IVERREL field of the control record as the identifier and ignore this field.
ICS03	P	X04	005	VR	Control version number The version number of the syntax used in the interchange and functional group control segments. Refer to the description of data element X04 in the standards manual for these version numbers. The LV or VR data type in the envelope definition signals the enveloper to use the IVERREL field of the control record as the version number and ignore this field.
ICS04	P	X05	002	ID	Sender ID qualifier A code for the method of designating the interchange sender in ICS05. Refer to the description of data element X05 in the standards manual for a list of these codes.
ICS05	P	X06	015	IS	Information sender ID The identification of the interchange sender. This is the interchange ID other trading partners use to route data to you. The IS data type in the envelope definition signals the enveloper to use the ISID field from the control record as the sender ID. If ISID is blank, ICS05 is used by default. If you do not provide a sender ID, the enveloper does not build an interchange envelope.

Table 6-6 (Page 2 of 5). ICS Envelope Data

Field Label	Data Source	Element ID	Length	Data Type	Description
ICS06	L	X05	002	ID	<p>Receiver ID qualifier</p> <p>A code for the method of designating the interchange receiver in ICS07. Refer to the description of data element X05 in the standards manual for a list of these codes. If the trading partner profile contains an interchange ID and interchange qualifier, the enveloper uses the interchange qualifier from the trading partner profile and ignores this field.</p>
ICS07	L	X08	015	IR	<p>Information receiver ID</p> <p>The identification of the interchange receiver. The IR data type in the envelope definition signals the enveloper to use the following sequence for supplying the ID:</p> <ul style="list-style-type: none"> If the default profile member is used: <ol style="list-style-type: none"> IRID field from the control record passed to the translator, if used Interchange ID from the trading partner profile Account number/user ID from the trading partner profile Value in ICS07 If the override profile member is used: <ol style="list-style-type: none"> IRID field from the control record passed to the translator, if used Value in ICS07 Interchange ID from the trading partner profile Account number/user ID from the trading partner profile
ICS08	L	X09	006	DT	<p>Interchange date</p> <p>The date when the sender prepared the interchange. The DT data type in the envelope definition signals the enveloper to use the system date and ignore this field.</p>
ICS09	L	X10	004	TM	<p>Interchange time</p> <p>The time (HHMM) at which the sender prepared the interchange. The TM data type in the envelope definition signals the enveloper to use the system time and ignore this field.</p>
ICS10	L	X11	009	CN	<p>Interchange control number</p> <p>A number used to reference the envelope. It must match ICE02. The CN data type in the envelope definition signals the enveloper to use the interchange control number from the trading partner profile and ignore this field.</p>
Interchange Trailer Segment (ICE)					
ICE01	L	X13	006	CT	<p>Number of included groups</p> <p>A control total of the number of functional groups in the interchange. The CT data type in the envelope definition signals the enveloper to use its internal counter and ignore this field.</p>

Customizing Envelope Standards

Table 6-6 (Page 3 of 5). ICS Envelope Data

Field Label	Data Source	Element ID	Length	Data Type	Description
ICE02	L	X11	009	CN	Interchange control number A number used to reference the envelope. It must match ICS10. The CN data type in the envelope definition signals the enveloper to use the same value that was used for ICS10.
Functional Group Header Segment (GS)					
GS01	L	479	002	FG	Functional group ID An identification of the type of transaction sets in the group. Refer to the description of data element 479 in the standards manual for a list of codes. The FG data type in the envelope definition signals the enveloper to use the functional group ID from the standard transaction definition and ignore this field.
GS02	L	142	015	AS	Application sender's code The name or code for a specific department in the sender's company. The AS data type in the envelope definition signals the enveloper to use other sources for the sender ID. The sources, in priority sequence, are: <ol style="list-style-type: none"> 1. GSID in the control record, if used 2. Application sender ID in the trading partner transaction 3. GS02 (this field) 4. The data format ID This sender ID can be different from ICS05.
GS03	L	124	015	AR	Application receiver's code The name or code for the specific department in the receiver's company that is to receive the group. The AR data type in the envelope definition signals the enveloper to use other sources for the recipient ID. The sources, in priority sequence, are: <ol style="list-style-type: none"> 1. GSID in the control record, if used 2. Application sender ID in the trading partner transaction 3. GS03 (this field) This receiver ID can be different from ICS07. Note: With 0 enveloping, no interchange (send usage envelope type = 0 or ISA06 is blank), DataInterchange uses the trading partner nickname.
GS04	L	029	008	DT	Data interchange date The date when the sender prepared the functional group. The DT data type in the envelope definition signals the enveloper to use the system date and ignore this field.
GS05	L	030	008	TM	Data interchange time The time (HHMM) at which the sender prepared the functional group. The TM data type in the envelope definition signals the enveloper to use the system time and ignore this field.

Table 6-6 (Page 4 of 5). ICS Envelope Data

Field Label	Data Source	Element ID	Length	Data Type	Description
GS06	L	028	009	CN	Functional group control number A number used to reference the envelope. It must match GE02. The CN data type in the envelope definition signals the enveloper to use the functional group control number from the trading partner profile and ignore this field. The IV data type should not be used for mapping this data element.
GS07	PS	455	002	ID	Responsible agency code A code used with GS08 to identify the agency that has control of the standard. Refer to the description of data element 455 in the standards manual for a list of these codes. Use the Envelope standards option on the main menu to change the data type of this field to AG if 'Responsible agency code' is used to qualify receive usages.
GS08	PS	480	012	ID	Version/release/industry ID A code for the version, release, and industry of the standard. Refer to the description of data element 480 in the standards manual for a list of these codes. You can override the version number by providing it in the TVER field of the control record passed to the translator. It maps to the VR data type in the envelope definition. Use the Envelope standards option on the main menu to change the data type of this field to VR or LV if 'Version' or 'Release' is used to qualify receive usages.
Functional Group Trailer Segment (GE)					
GE01	L	097	006	CT	Number of included sets A control total of the number of transaction sets in a functional group. The CT data type in the envelope definition signals the enveloper to use its internal counter and ignore this field.
GE02	L	028	009	CN	Functional group control number A number used to reference the envelope. It must match GS06. The CN data type in the envelope definition signals the enveloper to use the same value that was used for GS06.
Transaction Set Header Segment (ST)					
ST01	L	143	003	TC	Transaction set ID A code the controlling agency assigns to identify the transaction set type. See the description of data element 143 in the standards manual for a list of these IDs. The TC data type in the envelope definition signals the enveloper to use the standard transaction ID and ignore this field.

Customizing Envelope Standards

Table 6-6 (Page 5 of 5). ICS Envelope Data

Field Label	Data Source	Element ID	Length	Data Type	Description
ST02	L	329	009	CN	Transaction set control number A number used to reference the envelope. It must match SE02. The CN data type in the envelope definition signals the enveloper to use the transaction set control number from the trading partner profile and ignore this field. The IV data type in the envelope definition signals the enveloper to start at 1 and sequentially number the transactions in a group.
Transaction Set Header Segment (SE)					
SE01	L	096	010	CT	Number of included segments A control total of the number of segments in a transaction set. The CT data type in the envelope definition signals the enveloper to use its internal counter and ignore this field.
SE02	L	329	009	CN	Transaction set control number A number used to reference the envelope. It must match ST02. The CN or IV data type in the envelope definition signals the enveloper to use the same value that was used for ST02.

UNTDI Envelope Data

The T envelope profile contains data for UNTDI interchange envelopes. In the Data Source column:

P = Profile member is the primary source.

S = Supplied with product.

L = Logic used to fill field. See Description column.

Table 6-7 (Page 1 of 5). UNTDI Envelope Data

Field Label	Data Source	Element ID	Length	Data Type	Description
UNTDIKEY			008		Profile retrieval key The name of this member. For each standard you define with type T enveloping, you must provide a member whose name matches the standard ID. You can provide additional members with names that do not match a standard ID and use these names as overrides when mapping a trading partner transaction.
Interchange Header Segment (STX)					
STX01	PS	STDS1	004	AN	Syntax identifier The identification of the interchange syntax rules. You can override the syntax identifier by including it in the IVERREL field of the control record passed to the translator.
STX02	PS	STDS2	001	N0	Syntax version number The version number of the interchange syntax identified in STX01. You can override the syntax version by including it in the IVERREL field of the control record passed to the translator.

Table 6-7 (Page 2 of 5). UNTDI Envelope Data

Field Label	Data Source	Element ID	Length	Data Type	Description
STX03	P	FROM1	014	IS	<p>Sender code</p> <p>The code that identifies the sender. Either a sender code or a sender name (STX04) must be present. The IS data type in the envelope definition signals the enveloper to use the ISID field of the control record as the sender code. If ISID is blank, STX03 is used by default.</p>
STX04	P	FROM2	035	AN	<p>Sender name</p> <p>The name that identifies the sender. Either a sender name or a sender code (STX03) must be present. The IS data type in the envelope definition signals the enveloper to use the ISID field of the control record as the sender name. If ISID is blank, STX04 is used by default.</p> <p>Note: As supplied by DataInterchange, the IS data type is in STX03, not STX04.</p>
STX05	L	UNTO1	014	IR	<p>Recipient code</p> <p>The code that identifies the receiver. Either a recipient code or a recipient name (STX06) must be present. The IR data type in the envelope definition signals the enveloper to use the following sequence for supplying the code:</p> <ul style="list-style-type: none"> • If the default profile member is used: <ol style="list-style-type: none"> 1. IRID field from the control record passed to the translator, if used 2. Interchange ID from the trading partner profile 3. Account number/user ID from the trading partner profile 4. Value in STX05 • If the override profile member is used: <ol style="list-style-type: none"> 1. IRID field from the control record passed to the translator, if used 2. Value in STX05 3. Interchange ID from the trading partner profile 4. Account number/user ID from the trading partner profile

Customizing Envelope Standards

Table 6-7 (Page 3 of 5). UNTDI Envelope Data

Field Label	Data Source	Element ID	Length	Data Type	Description
STX06	L	UNTO2	035	AN	<p>Recipient name</p> <p>The name that identifies the receiver. Either a recipient name or a recipient code (STX05) must be present. The IR data type in the envelope definition signals the enveloper to use the following sequence for supplying the name:</p> <ul style="list-style-type: none"> • If the default profile member is used: <ol style="list-style-type: none"> 1. IRID field from the control record passed to the translator, if used 2. Interchange ID from the trading partner profile 3. Account number/user ID from the trading partner profile 4. Value in STX06 • If the override profile member is used: <ol style="list-style-type: none"> 1. IRID field from the control record passed to the translator, if used 2. Value in STX06 3. Interchange ID from the trading partner profile 4. Account number/user ID from the trading partner profile <p>Note: As supplied by DataInterchange, the IR data type is in STX05, not STX06.</p>
STX07	L	TRDT1	006	DT	<p>Date</p> <p>The date when the transmission was prepared. The DT data type in the envelope definition signals the enveloper to use the system date and ignore this field.</p>
STX08	L	TRDT2	006	TM	<p>Time HHMMSS</p> <p>The time at which the transmission was prepared. The TM data type in the envelope definition signals the enveloper to use the system time and ignore this field.</p>
STX09	L	SNRF	014	CN	<p>Interchange control reference</p> <p>A number used to reference the envelope. The CN data type in the envelope definition signals the enveloper to use the interchange control number from the trading partner profile and ignore this field.</p>
STX10	L	RCRF	014	PW	<p>Recipient's reference/password</p> <p>A password the recipient assigns, to be used as agreed upon by the trading partners. The PW data type in the envelope definition signals the enveloper to use the password from the trading partner profile and ignore this field. You can override the password by including one in the IPSW field of the control record passed to the translator.</p>

Table 6-7 (Page 4 of 5). UNTDI Envelope Data

Field Label	Data Source	Element ID	Length	Data Type	Description
STX11	L	APRF	014	AP	Application reference The sender's identification of the application area to which the transmission relates. The AP data type in the envelope definition signals the enveloper to use the IAPREF field of the control record as the application reference and ignore this field. If you provide a message user class in the requestor profile, it overrides this field when sending (not building) the envelope.
STX12	P	PRCD	001	ID	Transmission priority code The sender's code for processing priority, as agreed upon with the trading partner.
Interchange Trailer Segment (END)					
END01	L	NMST	005	TT	Total number of messages A control total of the number of messages in the transmission. The CT or TT data type in the envelope definition signals the enveloper to use its internal counter and ignore this field.
Message Header Segment (MHD)					
MHD01	P	MRSF	012	IV	Message reference A number used to reference the envelope. The CN data type in the envelope definition signals the enveloper to use the transaction set control number from the trading partner profile and ignore this field. The IV data type signals the enveloper to start at 1 and sequentially number the messages in the interchange envelope.
MHD02	P	TYPE1	006	TC	Message type A code the controlling agency assigns to identify the message type. The TC data type in the envelope definition signals the enveloper to use the standard transaction ID and ignore this field.
MHD03	PS	TYPE2	001	VR	Message version The version number for the message type. You can override the version number by providing it in the TVER field of the control record passed to the translator. It maps to the VR data type in the envelope definition.
MHD04	P	CARF	035	AN	Common access reference The key that relates all subsequent transfers of data to a common file. Trading partners can agree to using a key made up of components, but syntax separators cannot be used. DataInterchange does not use this field.
MHD05	P	STAT1	002	N0	Sequence of transfers A number the sender assigns to indicate that the message is an addition or change to a previous message. The first message in a sequence is number 1. DataInterchange does not use this field.

Customizing Envelope Standards

Table 6-7 (Page 5 of 5). UNTDI Envelope Data

Field Label	Data Source	Element ID	Length	Data Type	Description
MHD06	P	STAT2	001	A	First and last transfers A code for the first and last message in a sequence of the same type and relating to the same topic: C For creation. C must be present for the first transfer. F For final. F must be present for the last transfer of a sequence. DataInterchange does not use this field.

Message Trailer Segment (MTR)

MTR01	L	NOSG	004	CT	Number of segments in message A control total of the number of segments in a message. The CT data type in the envelope definition signals the enveloper to use its internal counter and ignore this field.
-------	---	------	-----	----	---

UCS Envelope Data

The U envelope profile contains data for UCS interchange envelopes. In the Data Source column:

P = Profile member is the primary source.

S = Supplied with product.

L = Logic used to fill field. See Description column.

Table 6-8 (Page 1 of 5). UCS Envelope Data

Field Label	Data Source	Element ID	Length	Data Type	Description
UCSKEY			008		Profile retrieval key The name of this member. For each standard you define with type U enveloping, you must provide a member whose name matches the standard ID. You can provide additional members with names that do not match a standard ID and use these names as overrides when mapping a trading partner transaction.

Interchange Header Segment (BG)

BG01	P	402	010	AN	Comm ID The identification of the transmitting company.
BG02	L	403	010	AN	Comm Password A password the receiver assigns, to be used as agreed upon by the trading partners. The PW data type in the envelope definition signals the enveloper to use the password from the trading partner profile and ignore this field. You can override the password by including one in the ISPW field of the control record passed to the translator.

Table 6-8 (Page 2 of 5). UCS Envelope Data

Field Label	Data Source	Element ID	Length	Data Type	Description
BG03	P	142	015	IS	<p>Application Sender ID</p> <p>The area code and phone number of the sender's modem, without blanks or punctuation. The IS or AS data type in the envelope definition signals the enveloper to use the ISID field of the control record as the sender's ID. If ISID is blank, BG03 is used by default.</p> <p>If you do not provide a sender ID, the enveloper does not build an interchange envelope.</p>
BG04	L	124	015	IR	<p>Application Receiver ID</p> <p>The area code and phone number of the receiver's modem, without blanks or punctuation. The IR or AR data type in the envelope definition signals the enveloper to use the following sequence for supplying the ID:</p> <ul style="list-style-type: none"> If the default profile member is used: <ol style="list-style-type: none"> IRID field from the control record passed to the translator, if used Interchange ID from the trading partner profile Account number/user ID from the trading partner profile Value in BG04 If the override profile member is used: <ol style="list-style-type: none"> IRID field from the control record passed to the translator, if used Value in BG04 Interchange ID from the trading partner profile Account number/user ID from the trading partner profile
BG05	L	029	006	DT	<p>Data Interchange Date</p> <p>The date when the sender prepared the interchange. The DT data type in the envelope definition signals the enveloper to use the system date and ignore this field.</p>
BG06	L	030	006	TM	<p>Data Interchange Time</p> <p>The time (HHMM) at which the sender prepared the interchange. The TM data type in the envelope definition signals the enveloper to use the system time and ignore this field.</p>
BG07	L	404	005	N0	<p>Transmission Control Number</p> <p>A number used to reference the envelope. It must match EG01. The CN data type in the envelope definition signals the enveloper to use the interchange control number from the trading partner profile and ignore this field.</p>
Interchange Trailer Segment (EG)					
EG01	L	404	005	N0	<p>Transmission Control Number</p> <p>A number used to reference the envelope. It must match BG07. The CN data type in the envelope definition signals the enveloper to use the same value that was used for BG07.</p>

Customizing Envelope Standards

Table 6-8 (Page 3 of 5). UCS Envelope Data

Field Label	Data Source	Element ID	Length	Data Type	Description
EG02	L	405	005	CT	<p>Number of included functional groups</p> <p>A control total of the number of functional groups in the interchange. The CT data type in the envelope definition signals the enveloper to use its internal counter and ignore this field.</p>
EG03	L	097	006	CT	<p>Number of included transaction sets</p> <p>A control total of the number of transaction sets in the interchange. The TT data type in the envelope definition signals the enveloper to use its internal counter and ignore this field.</p>
EG04	L	096	010	CT	<p>Number of included data segments</p> <p>A control total of the number of segments in the interchange. The TS data type in the envelope definition signals the enveloper to use its internal counter and ignore this field.</p>
Functional Group Header Segment (GS)					
GS01	L	479	002	FG	<p>Functional group ID</p> <p>An identification of the type of transaction sets in the group. The FG data type in the envelope definition signals the enveloper to use the functional group ID from the trading partner transaction and ignore this field.</p>
GS02	L	142	015	IS	<p>Application sender's code</p> <p>The name or code for a specific department in the sender's company. The AS or IS data type in the envelope definition signals the enveloper to use other sources for the sender ID. The sources, in priority sequence, are:</p> <ul style="list-style-type: none"> • GSID in the control record, if used • Application sender ID in the trading partner transaction • GS02 (this field) • Data format ID <p>This sender ID can be different from BG03.</p>
GS03	L	124	015	IR	<p>Application receiver's code</p> <p>The name or code for the specific department in the receiver's company that is to receive the interchange. The AR or IR data type in the envelope definition signals the enveloper to use other sources for the receiver ID. The sources, in priority sequence, are:</p> <ul style="list-style-type: none"> • GSID in the control record, if used • Application receiver ID in the trading partner transaction • GS03 (this field) <p>This receiver ID can be different from BG04.</p> <p>Note: With 0 enveloping (no interchange, or send usage envelope type = 0, or ISA06 is blank), DataInterchange uses the trading partner nickname.</p>

Table 6-8 (Page 4 of 5). UCS Envelope Data

Field Label	Data Source	Element ID	Length	Data Type	Description
GS04	L	029	008	DT	Data interchange date The date when the sender prepared the functional group. The DT data type in the envelope definition signals the enveloper to use the system date and ignore this field.
GS05	L	030	006	TM	Data interchange time The time (HHMMSS) at which the sender prepared the functional group. The TM data type in the envelope definition signals the enveloper to use the system time and ignore this field.
GS06	L	028	009	CN	Functional group control number A number used to reference the envelope. It must match GE02. The CN data type in the envelope definition signals the enveloper to use the functional group control number from the trading partner profile and ignore this field.
GS07	PS	455	002	ID	Responsible agency code A code for the agency that has control of the standard. It is used in conjunction with the version information. Use the Envelope standards option on the main menu to change the data type of this field to AG if 'Responsible agency code' is used to qualify receive usages.
GS08	PS	480	012	ID	Version/release/industry ID A code for the version, release, and industry of the standard. Refer to the data element dictionary for a definition of this code. You can override the version number by providing it in the TVER field of the control record passed to the translator. It maps to the VR and LV data types in the envelope definition. Use the Envelope standards option on the main menu to change the data type of this field to VR or LV if 'Version' or 'Release' is used to qualify receive usages.
Functional Group Trailer Segment (GE)					
GE01	L	097	006	CT	Number of included sets A control total of the number of transaction sets in a functional group. The CT data type in the envelope definition signals the enveloper to use its internal counter and ignore this field.
GE02	L	028	009	CN	Functional group control number A number used to reference the envelope. It must match GS06. The CN data type in the envelope definition signals the enveloper to use the same value that was used for GS06.

Customizing Envelope Standards

Table 6-8 (Page 5 of 5). UCS Envelope Data

Field Label	Data Source	Element ID	Length	Data Type	Description
Transaction Set Header Segment (ST)					
ST01	L	143	003	TC	Transaction set ID A code the controlling agency assigns to identify the transaction set type. See the data element dictionary for the standard for a list of these IDs. The TC data type in the envelope definition signals the enveloper to use the standard transaction ID from the trading partner transaction and ignore this field.
ST02	L	329	009	CN	Transaction set control number A number used to reference the envelope. It must match SE02. The CN data type in the envelope definition signals the enveloper to use the transaction set control number from the trading partner profile and ignore this field. The IV data type signals the enveloper to start at 1 and sequentially number the transactions in a group.
Transaction Set Trailer Segment (SE)					
SE01	L	096	010	CT	Number of included segments A control total of the number of segments in a transaction set. The CT data type in the envelope definition signals the enveloper to use its internal counter and ignore this field.
SE02	L	329	009	CN	Transaction set control number A number used to reference the envelope. It must match ST02. The CN or IV data type in the envelope definition signals the enveloper to use the same value that was used for ST02.

X12 Envelope Data

The X envelope profile contains data for X12 interchange envelopes. In the Data Source column:

P = Profile member is the primary source.

S = Supplied with product.

L = Logic used to fill field. See Description column.

Table 6-9 (Page 1 of 7). X12 Envelope Data

Field Label	Data Source	Element ID	Length	Data Type	Description
X12KEY			008		Profile retrieval key The name of this member. For each standard you define with type X enveloping, you must provide a member whose name matches the standard ID. You can provide additional members with names that do not match a standard ID and use these names as overrides when mapping a trading partner transaction.

Table 6-9 (Page 2 of 7). X12 Envelope Data

Field Label	Data Source	Element ID	Length	Data Type	Description						
Interchange Header Segment (ISA)											
ISA01	PS	I01	002	ID	Authorization information qualifier A code for the type of information in ISA02. Refer to the description of data element I01 in the standards manual for the acceptable codes.						
ISA02	P	I02	010	AN	Authorization information Information used to further identify or authorize the sender or the interchange data. The code in ISA01 defines the type of information.						
ISA03	SL	I03	002	ID	Security information qualifier A code for the type of information in ISA04. Valid values are: <table><tr><th>Code</th><th>Description</th></tr><tr><td>00</td><td>ISA04 is not meaningful.</td></tr><tr><td>01</td><td>ISA04 contains a password.</td></tr></table> If <i>Interchange send password</i> in the trading partner profile contains a password, the enveloper uses the value 01, even if ISA03 contains 00.	Code	Description	00	ISA04 is not meaningful.	01	ISA04 contains a password.
Code	Description										
00	ISA04 is not meaningful.										
01	ISA04 contains a password.										
ISA04	L	I04	010	PW	Security information Security information about the sender or interchange data. The code in ISA03 defines the type of information. The PW data type in the envelope definition signals the enveloper to use the password from the trading partner profile and ignore this field. You can override the password by including one in the ISPW field of the control record passed to the translator.						
ISA05	PS	I05	002	ID	Interchange ID qualifier A code for the method of designating the interchange sender in ISA06. Refer to the description of data element I05 in the standards manual for a list of these codes.						
ISA06	P	I06	015	IS	Interchange sender ID The identification of the interchange sender. This is the interchange ID your trading partners use to route data to you. The IS data type in the envelope definition signals the enveloper to use the ISID field of the control record as the sender ID. If ISID is blank, ISA06 is used by default. If you do not provide a sender ID, the enveloper does not build an interchange envelope.						
ISA07	L	I05	002	ID	Interchange ID qualifier A code for the method of designating the interchange receiver in ISA08. Refer to the description of data element I05 in the standards manual for a list of these codes. If the trading partner profile contains an interchange ID and interchange qualifier, the enveloper uses the interchange qualifier from the trading partner profile and ignores this field.						

Customizing Envelope Standards

Table 6-9 (Page 3 of 7). X12 Envelope Data

Field Label	Data Source	Element ID	Length	Data Type	Description				
ISA08	L	I07	015	IR	<p>Interchange receiver ID</p> <p>The identification of the interchange receiver. The IR data type in the envelope definition signals the enveloper to use the following sequence for supplying the ID:</p> <ul style="list-style-type: none">• If the default profile member is used:<ol style="list-style-type: none">1. IRID field from the control record passed to the translator, if used2. Interchange ID from the trading partner profile3. Account number/user ID from the trading partner profile4. Value in ISA08• If the override profile member is used:<ol style="list-style-type: none">1. IRID field from the control record passed to the translator, if used2. Value in ISA083. Interchange ID from the trading partner profile4. Account number/user ID from the trading partner profile				
ISA09	L	I08	006	DT	<p>Interchange date</p> <p>The date when the sender prepared the interchange. The DT data type in the envelope definition signals the enveloper to use the system date and ignore this field.</p>				
ISA10	L	I09	004	TM	<p>Interchange time</p> <p>The time (HHMM) at which the sender prepared the interchange. The TM data type in the envelope definition signals the enveloper to use the system time and ignore this field.</p>				
ISA11	PS	I10	001	ID	<p>Interchange standards ID</p> <p>A code for the agency that controls the interchange syntax:</p> <table><thead><tr><th>Code</th><th>Description</th></tr></thead><tbody><tr><td>U</td><td>US EDI community of ASC X12, TDCC, and UCS</td></tr></tbody></table> <p>The LV or VR data type in the envelope definition signals the enveloper to use the IVERREL field of the control record as the interchange standards ID and ignore this field.</p>	Code	Description	U	US EDI community of ASC X12, TDCC, and UCS
Code	Description								
U	US EDI community of ASC X12, TDCC, and UCS								
ISA12	PS	I11	005	ID	<p>Interchange version ID</p> <p>The version number of the syntax used in the interchange and functional group control segments. Refer to the description of data element I11 in the standards manual for these version numbers.</p> <p>The LV or VR data type in the envelope definition signals the enveloper to use the IVERREL field of the control record as the interchange version and ignore this field.</p>				

Table 6-9 (Page 4 of 7). X12 Envelope Data

Field Label	Data Source	Element ID	Length	Data Type	Description						
ISA13	L	I12	009	CN	Interchange control number A number used to reference the envelope. It must match IEA02. The CN data type in the envelope definition signals the enveloper to use the interchange control number from the trading partner profile and ignore this field.						
ISA14	P	I13	001	AN	Acknowledgment requested The sender's code for requesting an acknowledgment. Valid values are: <table><tr><th>Code</th><th>Description</th></tr><tr><td>0</td><td>Request no acknowledgment Note: Set to 0 when sending ISAs to your trading partner.</td></tr><tr><td>1</td><td>Request an acknowledgment that ISA and IEA segments were received and recognized Note: We do not recommend you change this value to 1. Please review “Interchange Acknowledgments (TA1 Segment)” on page 5-50 before adjusting this value.</td></tr></table>	Code	Description	0	Request no acknowledgment Note: Set to 0 when sending ISAs to your trading partner.	1	Request an acknowledgment that ISA and IEA segments were received and recognized Note: We do not recommend you change this value to 1. Please review “Interchange Acknowledgments (TA1 Segment)” on page 5-50 before adjusting this value.
Code	Description										
0	Request no acknowledgment Note: Set to 0 when sending ISAs to your trading partner.										
1	Request an acknowledgment that ISA and IEA segments were received and recognized Note: We do not recommend you change this value to 1. Please review “Interchange Acknowledgments (TA1 Segment)” on page 5-50 before adjusting this value.										
ISA15	L	I14	001	TI	Test indicator An indication that the interchange is for testing or production. Valid values are: <table><tr><th>Value</th><th>Description</th></tr><tr><td>T</td><td>For test data</td></tr><tr><td>P</td><td>For production data</td></tr></table> The TI data type in the envelope definition signals the enveloper to use the TESTIND field from the control record or the value of the RAWTEST keyword (for raw data) and ignore this field.	Value	Description	T	For test data	P	For production data
Value	Description										
T	For test data										
P	For production data										
ISA16	PS	I15	001	AN	Subelement separator The character that separates subelements of a composite element. Reserved for future expansion. The X12 envelope profile value is not used. Default values are: <ul style="list-style-type: none">• Taken from the X envelope standard definition.• Overridden via the Trading Partner Profile Subelement Delimiter field.						
Interchange Trailer Segment (IEA)											
IEA01	L	I16	005	CT	Number of included groups A control total of the number of functional groups in the interchange. The CT data type in the envelope definition signals the enveloper to use its internal counter and ignore this field.						

Customizing Envelope Standards

Table 6-9 (Page 5 of 7). X12 Envelope Data

Field Label	Data Source	Element ID	Length	Data Type	Description
IEA02	L	112	009	CN	Interchange control number A number used to reference the envelope. It must match ISA13. The CN data type in the envelope definition signals the enveloper to use the same value that was used for ISA13.
Functional Group Header Segment (GS)					
GS01	L	479	002	FG	Functional group ID An identification of the type of transaction sets in the group. Refer to the description of data element 479 in the standards manual for a list of codes. The FG data type in the envelope definition signals the enveloper to use the functional group ID from the trading partner transaction and ignore this field.
GS02	L	142	015	AS	Application sender's code The name or code for a specific department in the sender's company. The AS data type in the envelope definition signals the enveloper to use other sources for the sender ID. The sources, in priority sequence, are: <ul style="list-style-type: none"> • GSID in the control record, if used • Application sender ID in the trading partner transaction • GS02 (this field) • Data format ID This sender ID can be different from ISA06.
GS03	L	124	015	AR	Application receiver's code The name or code for the specific department in the receiver's company that is to receive the interchange. The AR data type in the envelope definition signals the enveloper to use other sources for the receiver ID. The sources, in priority sequence, are: <ul style="list-style-type: none"> • GSID in the control record, if used • Application receiver ID in the trading partner transaction • GS03 (this field) This receiver ID can be different from ISA08. Note: With 0 enveloping (no interchange, or send usage envelope type = 0, or ISA06 is blank), DataInterchange uses the trading partner nickname.
GS04	L	029	008	DT	Data interchange date The date when the sender prepared the functional group. The DT data type in the envelope definition signals the enveloper to use the system date and ignore this field.
GS05	L	030	006	TM	Data interchange time The time (HHMMSS) at which the sender prepared the functional group. The TM data type in the envelope definition signals the enveloper to use the system time and ignore this field.

Table 6-9 (Page 6 of 7). X12 Envelope Data

Field Label	Data Source	Element ID	Length	Data Type	Description						
GS06	L	028	009	CN	Functional group control number A number used to reference the envelope. It must match GE02. The CN data type in the envelope definition signals the enveloper to use the functional group control number from the trading partner profile and ignore this field.						
GS07	PS	455	002	ID	Responsible agency code A code used with GS08 to identify the agency that has control of the standard. Valid values are: <table><tr><th>Code</th><th>Description</th></tr><tr><td>T</td><td>For TDCC</td></tr><tr><td>X</td><td>For ASC X12</td></tr></table> Use the Envelope standards option on the main menu to change the data type of this field to AG if 'Responsible agency code' is used to qualify receive usages.	Code	Description	T	For TDCC	X	For ASC X12
Code	Description										
T	For TDCC										
X	For ASC X12										
GS08	PS	480	012	ID	Version/release/industry ID A code for the version, release, and industry of the standard. Refer to the description of data element 480 in the standards manual for a list of these codes. You can override the version number by providing it in the TVER field of the control record passed to the translator. It maps to the VR data type in the envelope definition. Use the Envelope standards option on the main menu to change the data type of this field to VR or LV if 'Version' or 'Release' is used to qualify receive usages.						
Functional Group Trailer Segment (GE)											
GE01	L	097	006	CT	Number of included sets A control total of the number of transaction sets in a functional group. The CT data type in the envelope definition signals the enveloper to use its internal counter and ignore this field.						
GE02	L	028	009	CN	Functional group control number A number used to reference the envelope. It must match GS06. The CN data type in the envelope definition signals the enveloper to use the same value that was used for GS06.						
Transaction Set Header Segment (ST)											
ST01	L	143	003	TC	Transaction set ID A code the controlling agency assigns to identify the transaction set type. See the description of data element 143 in the standards manual for a list of these IDs. The TC data type in the envelope definition signals the enveloper to use the standard transaction ID and ignore this field.						

Customizing Envelope Standards

Table 6-9 (Page 7 of 7). X12 Envelope Data

Field Label	Data Source	Element ID	Length	Data Type	Description
ST02	L	329	009	CN	Transaction set control number A number used to reference the envelope. It must match SE02. The CN data type in the envelope definition signals the enveloper to use the transaction set control number from the trading partner profile and ignore this field.
Transaction Set Trailer Segment (SE)					
SE01	L	096	010	CT	Number of included segments A control total of the number of segments in a transaction set. The CT data type in the envelope definition signals the enveloper to use its internal counter and ignore this field.
SE02	L	329	009	CN	Transaction set control number A number used to reference the envelope. It must match ST02. The CN data type in the envelope definition signals the enveloper to use the same value that was used for ST02.

Building the Enveloping Segments

The source of data for the enveloping segments is illustrated in the following figures:

This Figure:	Shows the Building of the:
Figure 6-4 on page 6-61	ISA segment (override)
Figure 6-5 on page 6-62	ISA segment (default)
Figure 6-6 on page 6-63	GS segment
Figure 6-7 on page 6-64	ST segment
Figure 6-8 on page 6-65	UNB segment (override)
Figure 6-9 on page 6-66	UNB segment (default)
Figure 6-10 on page 6-67	UNG segment
Figure 6-11 on page 6-68	UNH segment

These diagrams show the various sources that are used in constructing enveloping segments (ISA, GS, ST, UNB, UNG, UNH). For several fields, data can originate from more than one source. When this occurs, all sources are shown in priority sequence. For example, in Figure 6-4 on page 6-61, the ISA08 field shows four sources:

1. The IRID field from the C record in the application data file
2. The ISA08 field from the appropriate I profile member
3. The Interchange ID field from the trading partner profile
4. The account number and User ID fields from the trading partner file

The sources are examined in priority sequence until a source with non-blank data is found.

The use of data, other than delimiters, from any place other than the envelope profile member depends on the data type for the field as defined in the envelope standard. For any field with data type ID, the value for that field is always taken from the envelope profile member, never from any other source. The assumed data types for fields are shown at the far right of the diagram.

Notes:

1. The only difference between Figure 6-4 on page 6-61 and Figure 6-5 on page 6-62 is the order that fields are searched to construct the receiver-ID value (ISA08).
2. The only difference between Figure 6-8 on page 6-65 and Figure 6-9 on page 6-66 is the order that fields are searched to construct the receiver-ID value (0010).

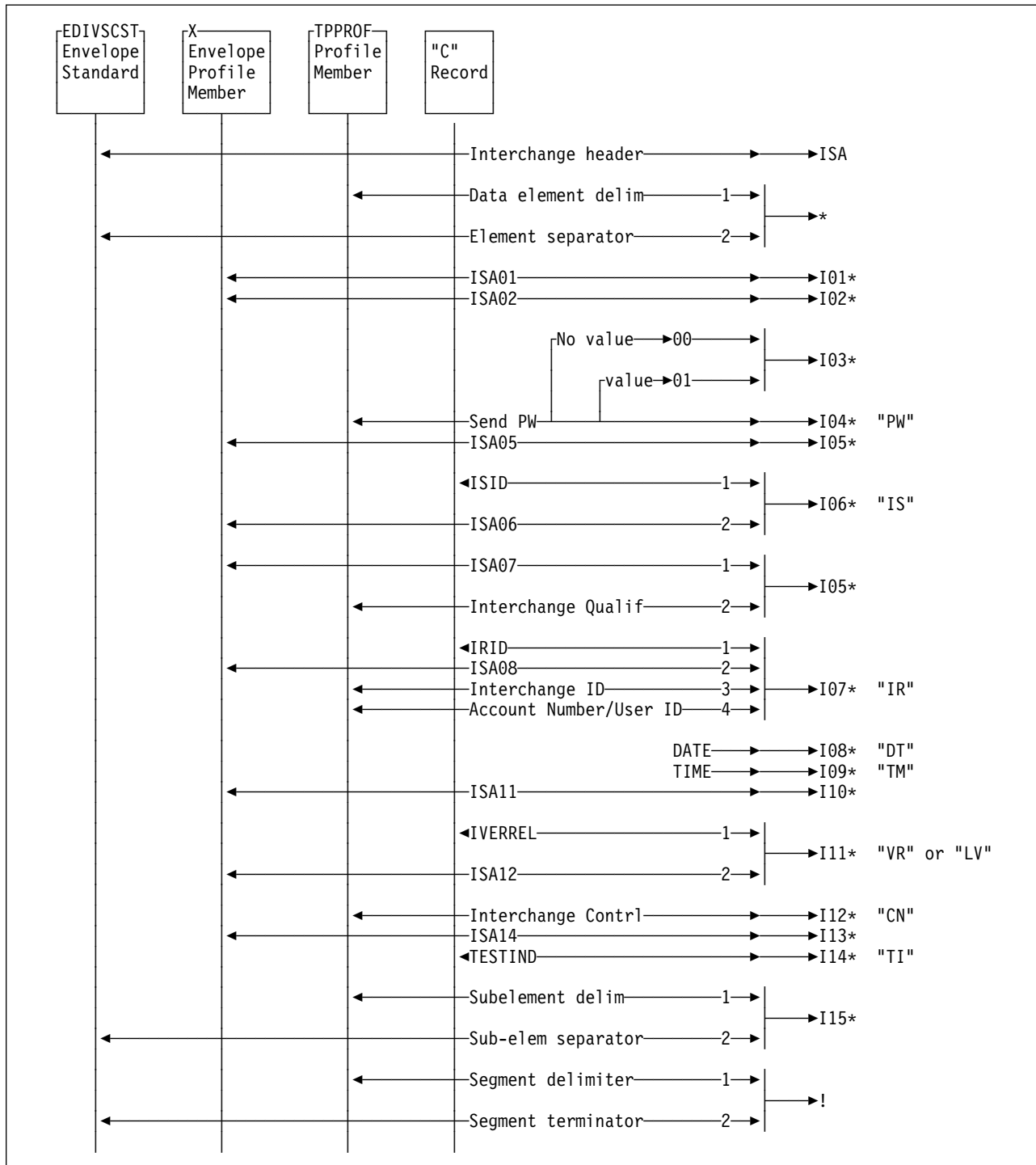


Figure 6-4. Building the ISA segment (Override). Sources for data when building the ISA segment and using a "Standard profile member name" override.

Customizing Envelope Standards

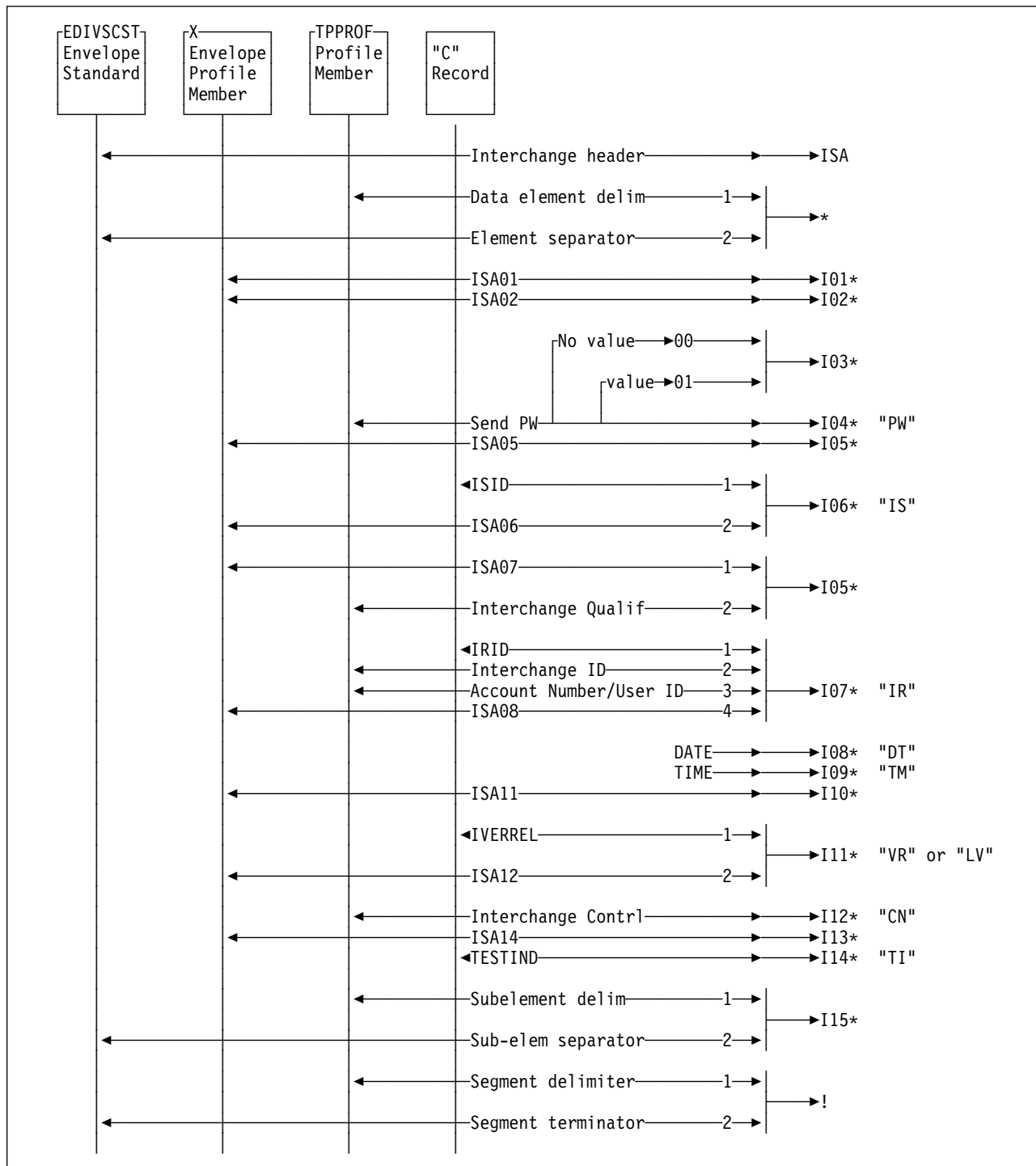


Figure 6-5. Building the ISA Segment (Default). Sources for data when building the ISA segment and using the default "Standard profile member name."

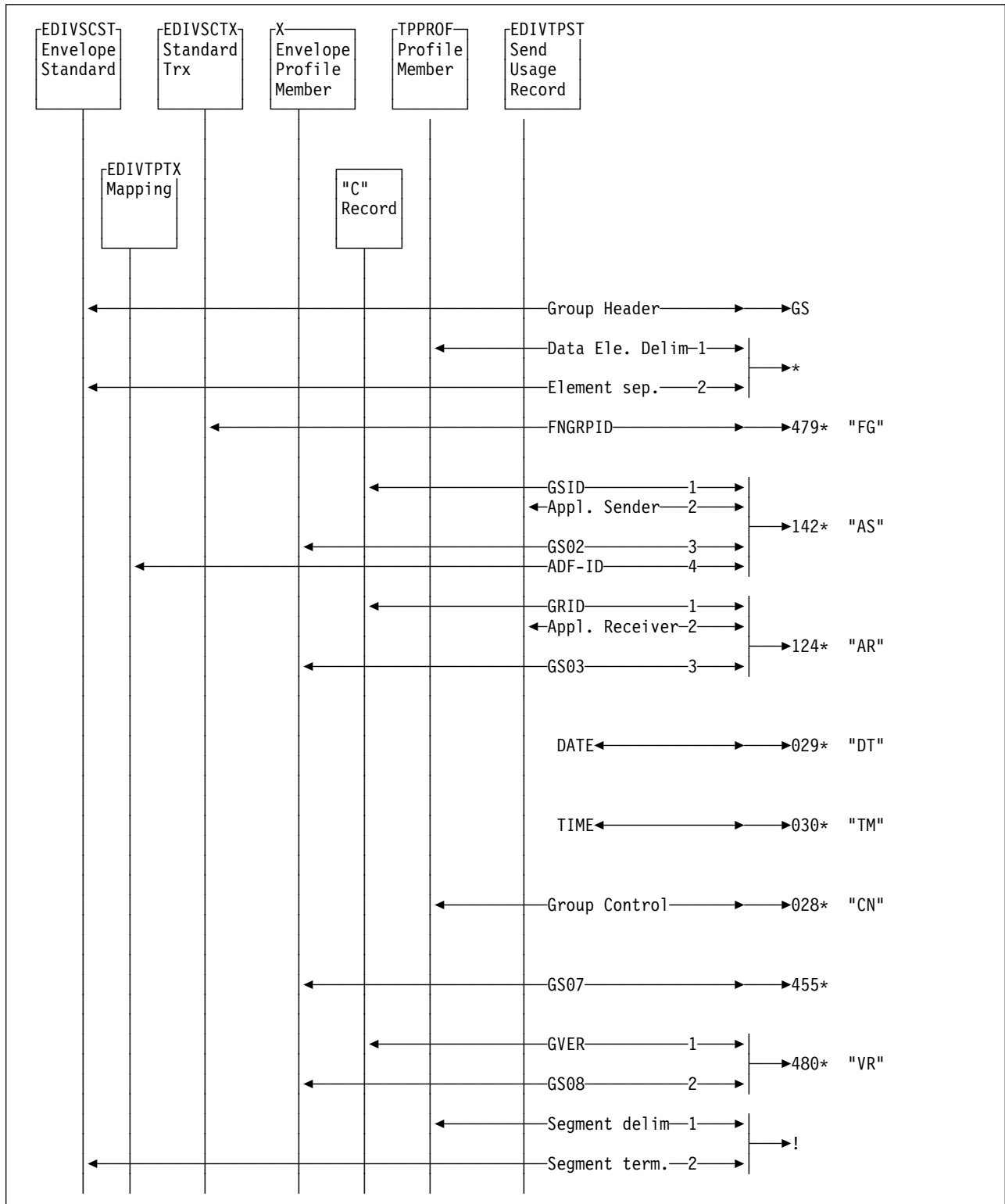


Figure 6-6. Building the GS Segment. Sources for data when building the GS segment.

Customizing Envelope Standards

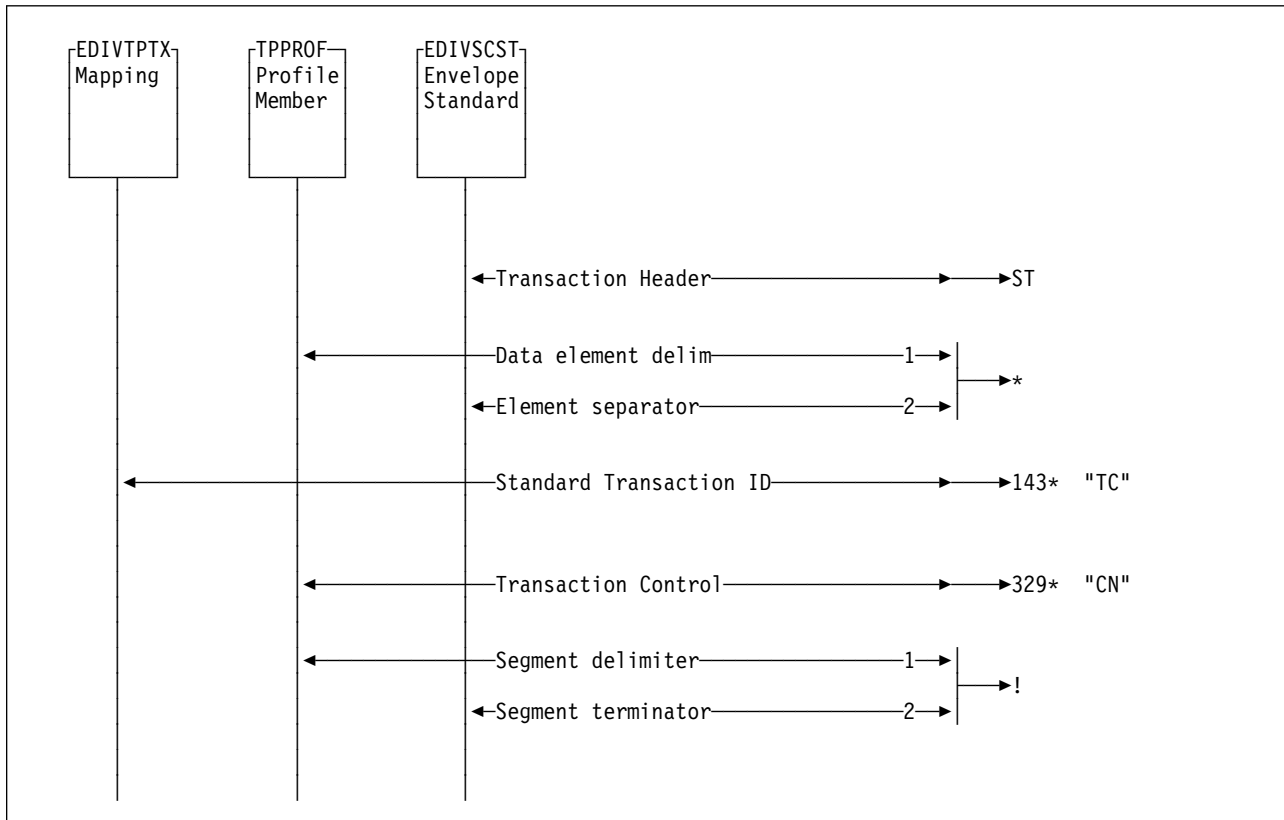


Figure 6-7. Building the ST Segment. Sources for data when building the ST segment.

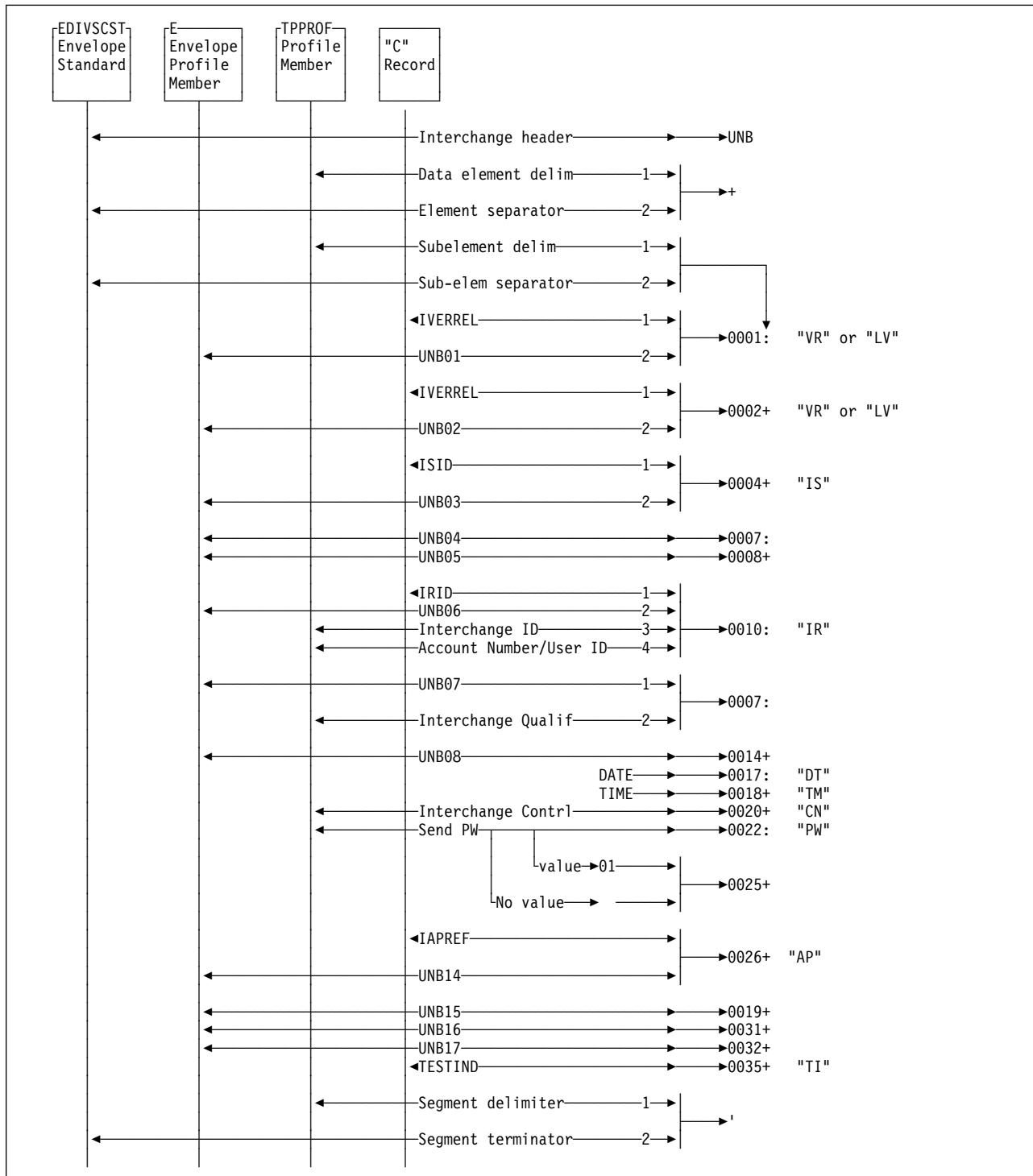


Figure 6-8. Building the UNB Segment (Override). Sources for data when building the UNB segment and using a "Standard profile member name" override.

Customizing Envelope Standards

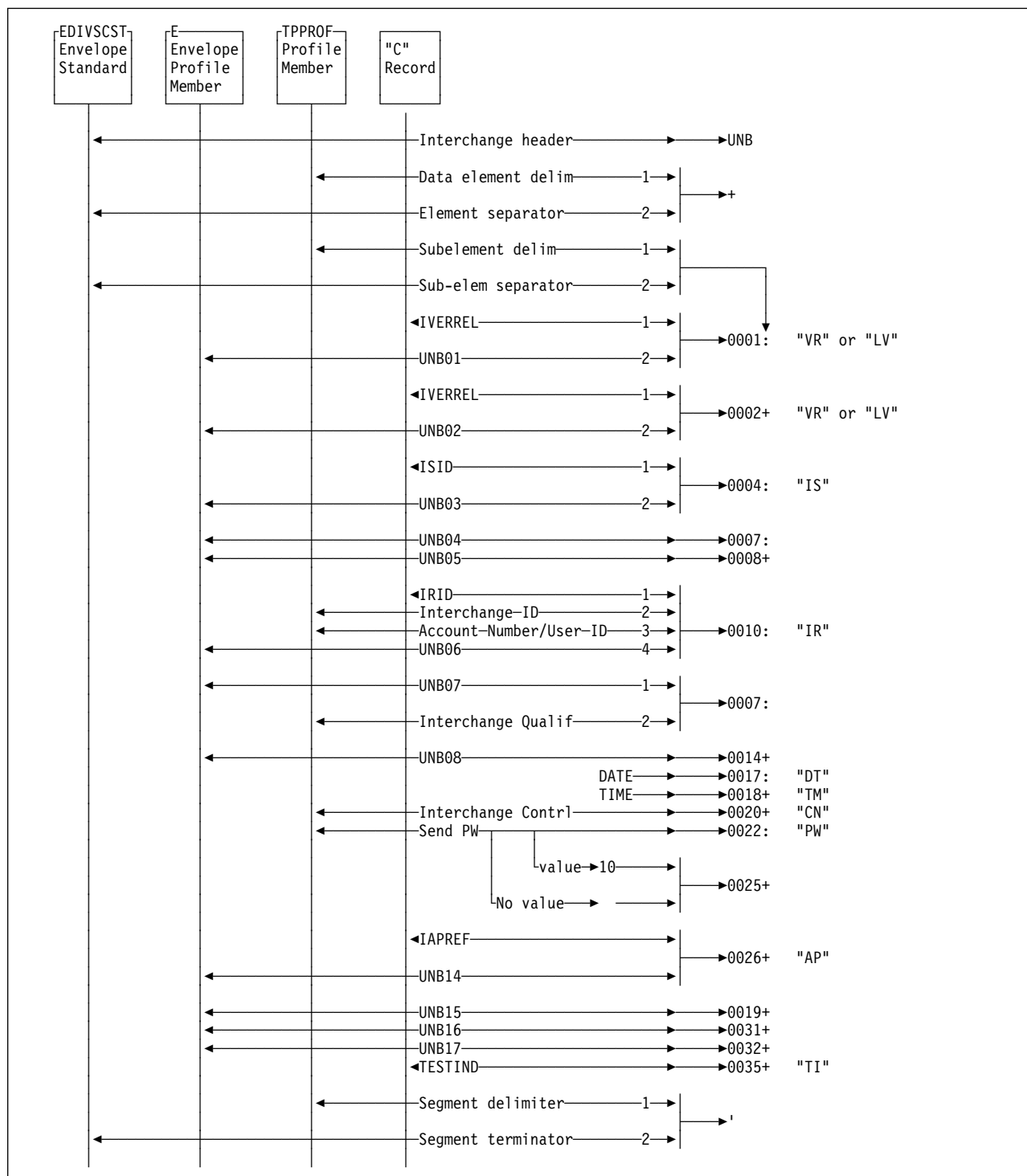


Figure 6-9. Building the UNB Segment (Default). Sources for data when building the UNB segment and using the default "Standard profile member name."

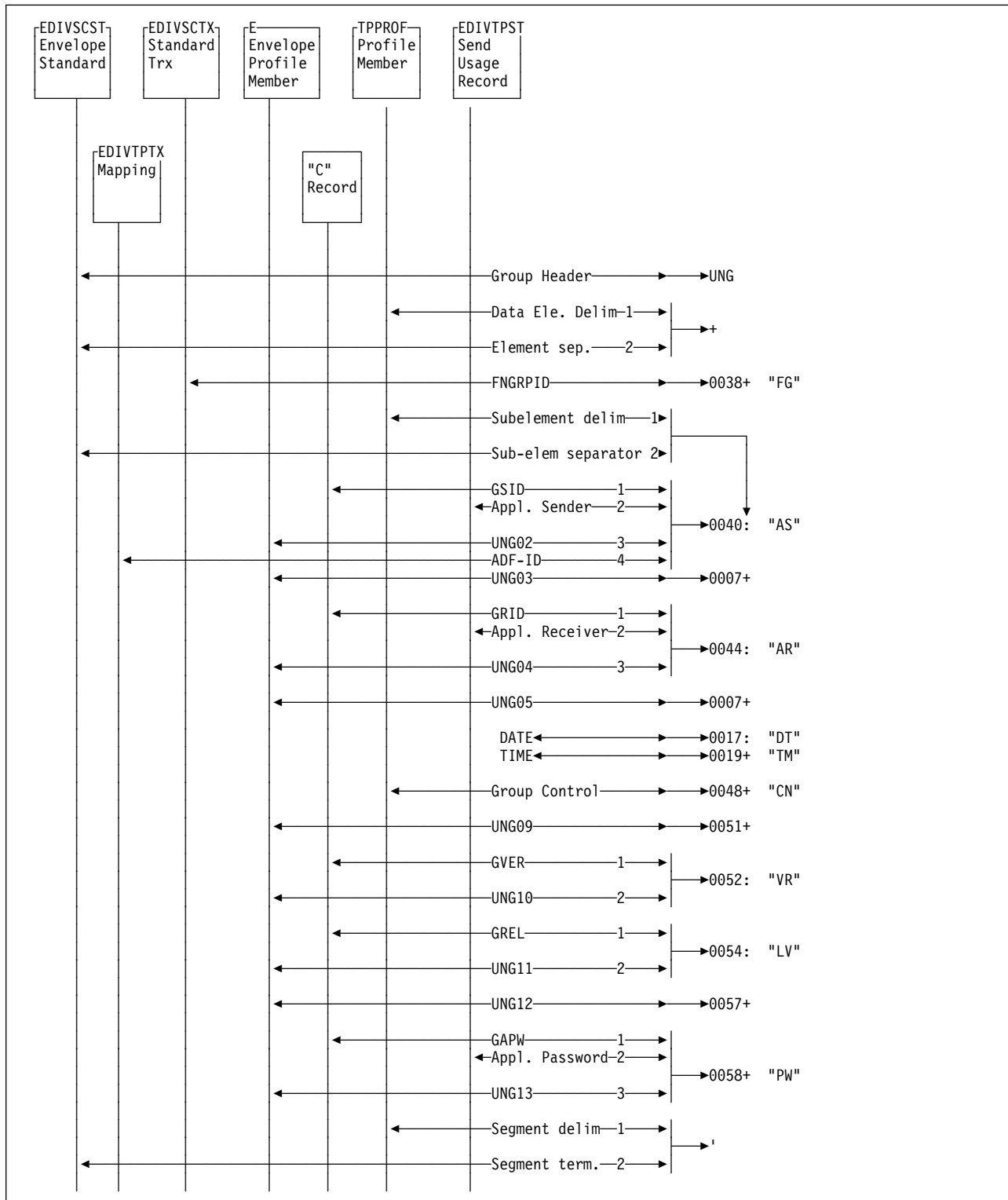


Figure 6-10. Building the UNG Segment. Sources for data when building the UNG segment.

Customizing Envelope Standards

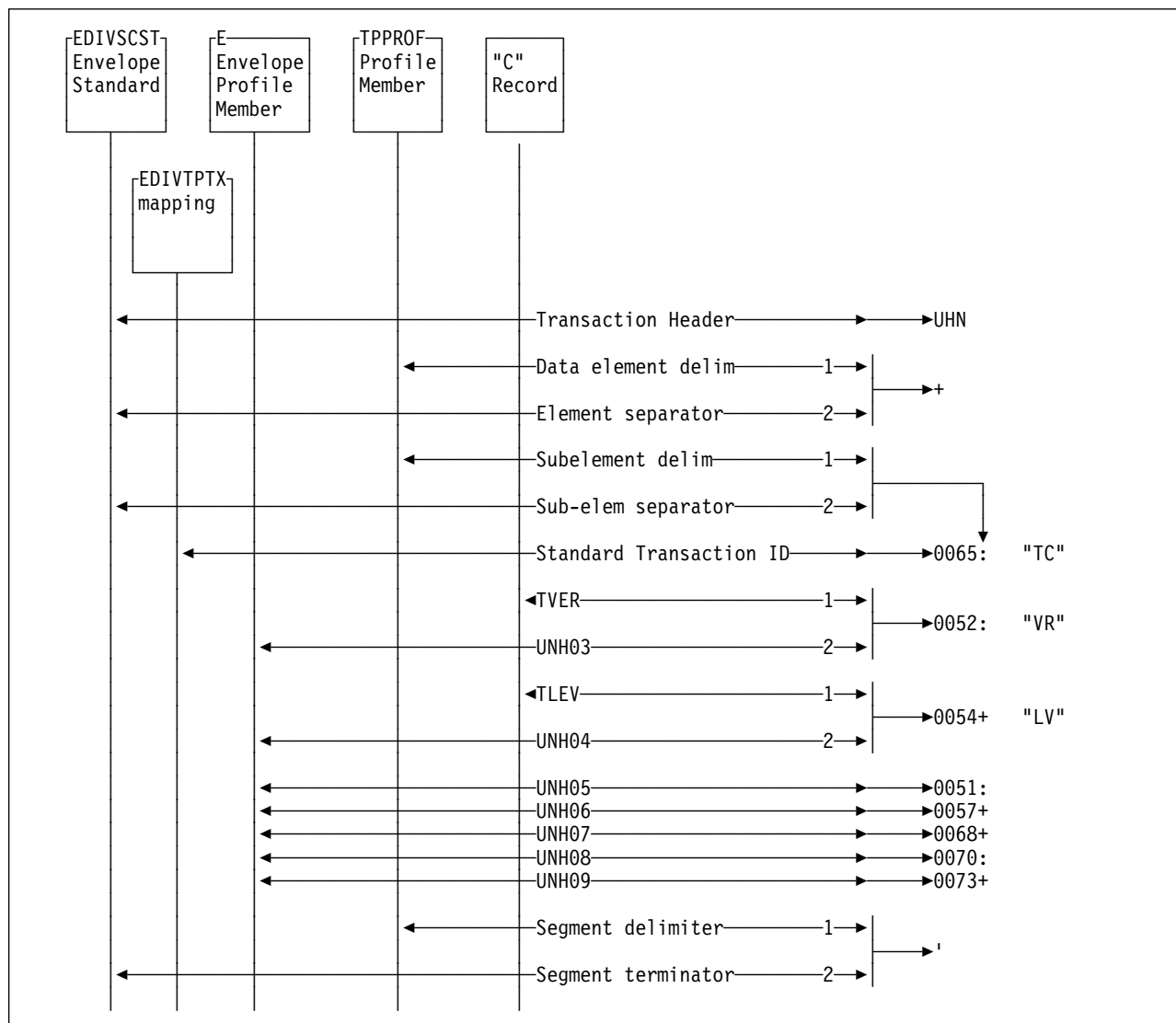


Figure 6-11. Building the UNH Segment. Sources for data when building the UNH segment.

Chapter 7. Defining and Working with Your Application Data

Your Application Data	7-1
Record Formats Used by DataInterchange	7-1
Understanding Application Data Structures	7-3
DataInterchange Application Data Format	7-4
Application Data Format Worksheet	7-4
Adding an Application Data Format	7-7
Data Types for Application Data Formats	7-17
Reformatting Data into Control (C) and Data (D) Records	7-20
One D Record for Each Transaction	7-20
One D Record for Each Data Structure	7-21
Variable D Records	7-22
Changing Your Application Data Format after Mapping	7-23
Creating an Application Data Format for an Empty Transaction	7-23
Translating Application Data to Application Data (Fixed-to-Fixed)	7-23
Setting up Fixed-to-Fixed Translation	7-24
FIXED Standards without an Associated ADF	7-26
Blank Application Data Format Worksheet	7-26

Chapter 7. Defining and Working With Your Application Data

This chapter provides information about defining your application data to DataInterchange. It describes the different types of record layouts used by DataInterchange and helps you determine which layout best meets your needs. This chapter assumes that you are familiar with your application data and the EDI standard transaction set you are using. You must define the layout of your application data to DataInterchange before it can be mapped to an EDI standard. This chapter helps you:

- Decide on a record structure to define to DataInterchange
- Define your application data layout to DataInterchange
- Work with your application data format

An Application Data Format (ADF) is used to define your application data to DataInterchange. You normally create a separate ADF for each business document you exchange using EDI because the format is different for each type of business document. For example, you should have an ADF for purchase orders you send, and another ADF for invoices you receive.

Your Application Data

You must understand how your application data is structured to effectively define it to DataInterchange. If you do not already have a copy of your application's record layouts, you need to get them. The record layouts can be from your application program code listing or any other documentation that clearly shows the starting and ending position of each field in the record layout. It should also show the physical attributes and contents of each field, the position of each field in the record, and the relationship between the records. This information is necessary to determine if you need to modify your application data before using DataInterchange.

Record Formats Used by DataInterchange

DataInterchange can use two types of record formats. The first type is referred to as raw data. Each record in raw data contains a unique record identifier (ID) specifying the type of information contained in the record. DataInterchange requires that the record ID is always found in the same starting position and is always the same length. Figure 7-1 shows application data with a unique record identifier in the first three bytes. In this example, HDR tells you the record is a header record, ITM is a line item record, and so on.

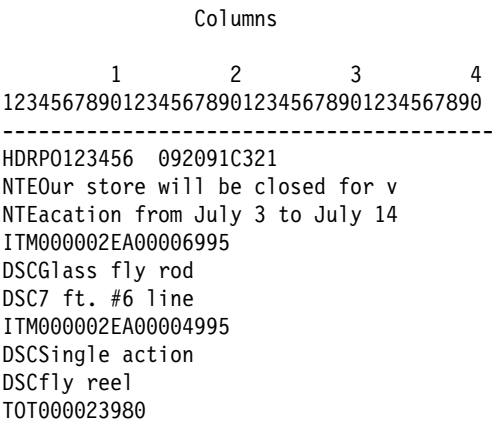


Figure 7-1. Sample Application Data with Record Identifier

Defining Your Application Data

The second type of record structure is the C and D record (Control and Data). C and D records are used when no record identifier exists clearly specifying the type of information contained in the record. Figure 7-2 and Figure 7-3 show the two possible formats for C and D records with sample application data. Figure 7-2 shows sample data with either a C or D in the first byte of each record and that the D is followed by 16 bytes equal to the structure name which is followed by the application data for that structure. The Z indicates the end of the transaction. For more information on Z records, see page 128 in the *DataInterchange Programmer's Reference*. Figure 7-3 shows another example of C and D records. In this case the data consists of a single C and a single D record. This format may only be used when the application data format defines a single structure which is passed separately, in which case the structure name is not necessary and the application data may immediately follow the D record ID.

Note: The D record only shows the first 99 positions.

```
Columns
      1      2      3      4
12345678901234567890123456789012345678901234567
-----
CSPSTT16 AFTSU09 NY
DPOHDR      P0123456 092091C321
DPONOTE      Our store will be closed for v
DPONOTE      acation from July 3 to July 14
DPOLINITEM    000002EA00006995
DPODESC      Glass fly rod
DPODESC      7 ft. #6 line
DPOLINEITEM    000002EA00004995
DPODESC      Single action
DPODESC      fly reel
DPOTOT      000023980
Z
CSPSTT16 AFTSU09 NY
```

Figure 7-2. Sample Application Data with C and D Records

```
Columns
      1      2      3      4      5      6      7      8      9
123456789012345678901234567890123456789012345678901234567890123456789
-----
CSPSTT16 AFTSU09 NY
DP0123456 092091C321          Our store will be closed for vacation from July 3 to July 14.    ...
Z
CSPSTT16 AFTSU09 NY
```

Figure 7-3. Reformatted Data

Figure 7-4 on page 7-3 shows sample application data that does not have a record identifier and is not formatted in C and D records. When your data has no record identifier for DataInterchange to associate with each record, you have two choices. You can modify your application data file to contain a record identifier or you can modify it to use C and D records. Each type has its own advantages.

The advantage of raw data is:

- You can use data records from the application without modification (as long as they contain an identifiable record ID).

The advantages of C and D records are:

- You can use multiple formats within a single file.
- You can provide overrides for fields within service segments (ISA, GS, UNB, UNH, and so on).

See “Reformatting Data into Control (C) and Data (D) Records” on page 7-20 and the *DataInterchange Programmer’s Reference* if you need more information.

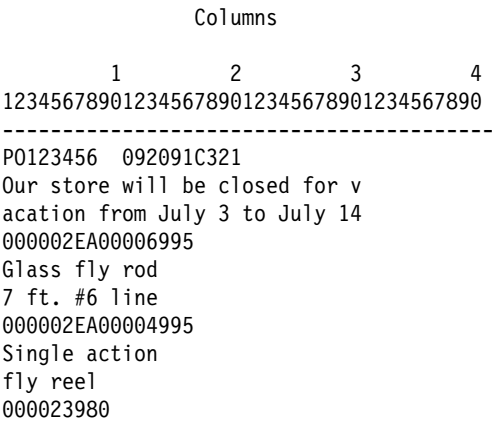


Figure 7-4. Sample Application Data

Understanding Application Data Structures

A data structure is a grouping of related data fields, such as the fields that make up a line item or the SHIP TO address of a purchase order.

DataInterchange needs information about the structure of your data, such as:

- Which data fields in your application data form structures
- The order in which structures appear
- The number of times the structure occurs

DataInterchange uses data structures to:

- Identify the data you pass

In some cases, such as when using multiple D records per transaction, the structure name is passed with the data identifying the structure to DataInterchange. In other cases, DataInterchange derives the structure name from the record ID.

DataInterchange uses the structure name to retrieve the characteristics of the application data. Your data format must, therefore, define a structure for each data record that is separately passed to the translator.

- Show relationships in the data

To get the correct results during translation, your data format must show how data structures relate to each other. This is accomplished using the ST data type to build a hierarchy of data structures. When you finish defining your structures, you can use the Data Structure Hierarchy panel (TD06) to verify the relationship of the data. The following panel illustrates this.

Defining Your Application Data

Add Copy Delete Update View		
TD01	Data Formats	1 to 2 of 2
A Data format ID		
— GENINV		
S POSAMPLE		

TD06	Data Structures Hierarchy	1 to 5 of 5
Data format ID : POSAMPLE		
Description : Sample purchase order		
Action	Structure	
—	POBASE	
—	SHIPTO	
—	DETAIL	
—	DETAILDESC	
—	TOTAL	

In this example, each data structure is independent of the other data structures, except for DETAILDESC which has a logical relationship with DETAIL. Whether you pass these structures separately or together in the application data does not matter. What is important is the relationship between the two be conveyed to DataInterchange. The relationship can be stated as:

For each occurrence of DETAIL, there can be one or more occurrences of DETAILDESC.

If this relationship is not established, DataInterchange has no way of associating a detail description to the detail line item.

- Show occurrences

You can think of some data structures as domains that supply information to the repeating segments and loops in a standard transaction. If all occurrences of a repeating segment or a loop are mapped to the same repeating area of the data format, there must be a data structure that sets the domain for mapping the repeating segment or loop.

A typical purchase order can be used to illustrate data structure concepts as they apply to DataInterchange. This purchase order might contain three sections, such as the HEADER, DETAIL, and TOTAL sections that are included in the X12 850 transaction set that we use to illustrate more concepts later in this chapter. Within the HEADER section, the typical purchase order might contain a SHIP TO structure, which would include information about where the merchandise should be shipped. The DETAIL section might include a structure which would consist of information necessary to process an order. To order more than one item, this structure would be repeated for each line item ordered. The TOTAL section might consist of one structure that summarizes the total.

DataInterchange Application Data Format

The ADF identifies the fields and record structures in your application data. You must define each field in your application record to DataInterchange even if that field does not contain data that is passed to your trading partner.

Application Data Format Worksheet

Creating an ADF worksheet before you begin defining your data to DataInterchange will save you time. A blank sample ADF worksheet is provided in “Blank Application Data Format Worksheet” on page 7-26. The worksheet helps you to analyze how your application data maps to the EDI standard transaction set you are using. It also shows you where special handling may be required.

Before you create the ADF worksheet, get a copy of your application data layout. For the sample purchase order, use Figure 7-5 on page 7-6, which illustrates four record types that might comprise a purchase order.

Later when you map your data, you will need to understand your trading partner's agreements.

Creating the ADF worksheet looks more difficult than it really is, so take it step by step. Using the sample working storage record definition (see Figure 7-5 on page 7-6), create your sample ADF worksheet as follows:

1. List your record names (COBOL structure names) and application field names in the first column. This column is titled **Application Field Name** in the sample worksheet. Indenting the field names under the record description makes it easier to read and gives a clear picture of how the data is structured. (See Table 7-1 on page 7-6.)
2. Assign an appropriate structure name to each record description in the second column. All structure and field names must be unique within the ADF. This column is titled **ADF Structure / ADF Field** in the sample worksheet. Using the name of the record description as a prefix, name the fields included in each record. This helps to keep the ADF descriptions unique; however, you can use any naming convention, as long as all ADF structure names are unique.
3. Identify the type of data for each item you have listed in the **Application Field Name**. Enter this information in the column titled Type. For a full list of valid data types, see Table 7-2 on page 7-17.
4. Write the length of each application field in the fourth column. This column is titled Length.
5. In the fifth column, specify how many times a structure may occur (max use). This column is titled Occurrences. The sample application data only allows one SHIP TO address and up to 1000 purchase order line items. Although you only want to order two items, indicate 1000 because you are defining the application data format and not the specific purchase order example. Likewise, indicate 1 for the SHIP TO address because the application only supports one SHIP TO address for each purchase order.
6. Most of the structures will be passed to DataInterchange separately. For example, DataInterchange reads a record and then passes it to the translator. However, sometimes a structure is physically part of another structure. Look at the Detail Description record name in the sample record definition. It is part of the Detail structure and will not be passed separately. Specify a Y if the structure is passed separately, or an N if it is not.

When you pass information to DataInterchange, you can pass structures individually or grouped together. When structures are grouped together, you must account for every occurrence of a structure. If you have a small group of structures that remain constant, you can group them together and pass them as a single structure. For example, if you have a structure called ADDRESS that consists of records called NAME, CITYSTATEZIP, and PHONE, you can group this information together and pass it to DataInterchange as a single structure instead of three structures.

However, suppose that you have a structure called LINEITEM that could occur up to 1000 times. You should pass this type of structure separately. You do not want to have to pad the application data with 998 blank occurrences of LINEITEM just to pass 2 valid occurrences. The raw data format and the C and D record format both support passing single structures and passing grouped structures.

Defining Your Application Data

```

01 SHIPTO.
05 RECORD-IDENTIFIER.
10 PURCHASE-ORDER-NUMBER PIC X(08).
10 RECORD-ID PIC X(02).
05 COMPANY-NAME PIC X(30).
05 COMPANY-DUNS PIC X(10).
05 COMPANY-ADDRESS.
10 STREET PIC X(30).
10 CITY PIC X(15).
10 STATE PIC X(02).
10 ZIPCODE PIC 9(09).
05 COMPANY-PHONE PIC 9(10).

01 DETAIL.
05 RECORD-IDENTIFIER.
10 PURCHASE-ORDER-NUMBER PIC X(08).
10 RECORD-ID PIC X(02).
05 ITEM-NUMBER PIC X(10).
05 ORDER-QUANTITY PIC 9(09).
05 ORDER-UNITS PIC X(01).
05 UNIT-PRICE PIC 9(09)V99.
05 UNIT-DISCOUNT PIC 9(09)V99.
05 EXTENSION PIC 9(09)V99.

05 DETAIL-DESCRIPTION OCCURS 3 TIMES.
10 RECORD-IDENTIFIER.
15 PURCHASE-ORDER-NUMBER PIC X(08).
15 RECORD-ID PIC X(02).
10 DESCRIPTION PIC X(30).

01 TOTAL.
05 RECORD-IDENTIFIER.
10 PURCHASE-ORDER-NUMBER PIC X(08).
10 RECORD-ID PIC X(02).
05 TOTAL-AMOUNT PIC 9(09)V99.

```

Figure 7-5. Sample Working Storage Record Definition of a Purchase Order

Table 7-1. Application Data Worksheet for Sample Purchase Order

Application Field Name	ADF Structure / ADF Field	Type	Length	Occurrences	Passed Separately
SHIPTO	SHIPTO	ST		1	Y
PURCHASE-ORDER-NUMBER	SHPONUMBER	CH	8		
RECORD-ID	SHRECID	CH	2		
COMPANY-NAME	SHNAME	CH	30		
COMPANY-DUNS	SHDUNS	CH	10		
STREET	SHSTREET	CH	30		
CITY	SHCITY	CH	15		
STATE	SHSTATE	CH	2		
ZIPCODE	SHZIP	N0	9		
COMPANY-PHONE	SHPHONE	N0	10		
DETAIL	DETAIL	ST		1000	Y
PURCHASE-ORDER-NUMBER	DLPONUMBER	CH	8		
RECORD-ID	DLRECID	CH	2		
ITEM-NUMBER	DLITMNUMB	CH	10		
ORDER-QUANTITY	DLQTY	N0	9		
ORDER-UNITS	DLUNITS	CH	1		
UNIT-PRICE	DLUPRICE	N2	9		
UNIT-DISCOUNT	DLUDISC	N2	9		
EXTENSION	DLEXT	N2	9		
DETAILDESC	DETAILDESC	ST		3	N
PURCHASE-ORDER-NUMBER	DDPONUMBER	CH	8		
RECORD-ID	DDRECID	CH	2		
DESCRIPTION	DDDESC	CH	30		
TOTAL-LINE	TOTAL	ST		1	Y
PURCHASE-ORDER-NUMBER	TLPONUMBER	CH	8		
RECORD-ID	TLRECID	CH	2		
TOTAL-AMOUNT	TLAMOUNT	N2	9		

When you have completed your ADF worksheet, you are ready to begin defining your data to DataInterchange.

Adding an Application Data Format

The following steps describe how to add an ADF.

1. From the Administrator's Menu, select **Application Data Formats**. The Data Formats panel (TD01) is displayed.
2. Type **a** in the action column next to any item, and press Enter.

Add Copy Delete List Print Update View Where used Structures			
cReate standard			
<hr/>			
TD01	Data Formats		1 to 1 of 1
A	Data format ID	Description	
a	GENINV	General use invoice - Send	

The Add Data Format panel (TD02) is displayed. Use this panel to provide general information about your transaction.

The following screen assumes that you are using the raw data format. If you are using C and D records, you do not have to fill in the following fields:

- Record ID position
- Record ID length
- Record ID data type
- Beginning structure name
- Ending structure name
- Trading partner ID field

TD02

Add Data Format

Data format ID

POSAMPLE

Description

Sample purchase_order

Base structure name . . .

POBASE

Application file name . .

Application file type . .

+

Record ID position

9

Record ID length

2

Record ID data type

ch

Beginning structure name

SHIPTO

Ending structure name . .

Trading partner ID field

SHDUNS

Generic routing code field

Interchange sender fields .

Qualifier

ID

Interchange receiver flds.

Qualifier

ID

Application TP ID field . .

EDI TP field

3. Complete the fields as follows, and press Enter.

Defining Your Application Data

In this field:

Data format ID

Enter:

The name you give this data format. This field uniquely identifies each data format to DataInterchange and is used for mapping. In the example, POSAMPLE is the ID for a data format that describes a sample purchase order.

You can use any combination of characters in the ALPHANUM table, up to 16 characters.

Note: If generic routing code field is defined or you choose to use generic usage, the **Data format ID** field is limited to 8 characters. A translation table is required to use generic routing code field or generic usage. For more information, see the discussion of the Generic routing code field description on page 7-11, and the sections “Defining Generic Send Usages” on page 9-85 and “Defining Generic Receive Usages” on page 9-96.

Description

A phrase that describes the data format, such as Sample purchase order.

You can use any character data up to 50 characters. Embedded blanks are allowed.

Base structure name

The name of the highest level structure in the data format. The base structure contains the inner structures and fields that compose the entire data format. In the example, POBASE is the name of the base structure of a purchase order.

You can use any combination of characters in the ALPHANUM table, up to 16 characters. Embedded blanks are not allowed. See “ALPHANUM and CHARSET Tables” on page 8-2 for more information.

Application file name

The name of the file that stores incoming transactions described by this data format. This field is used for transactions you receive, but not for transactions you send. You can override this entry while defining usages for specific trading partners. By specifying different files, you can route incoming transactions to different departments or areas of your business.

If this application data format is going to be the basis for a fixed standard (the target of a cReate standard action), then the value specified in the Application file name field will be the name given to the standard that is created from this format. Also when data is translated to this format (by a Fixed-to-Fixed mapping against the standard), the application file name identifies the file to which the application data should be written.

For MVS, this field specifies a data definition name (ddname), or the name of a DataInterchange MQSeries Queue profile member.

For CICS, this field specifies where the data is stored or how the data is to be processed. If this field indicates where the data is stored, it can represent either a ddname for a VSAM entry sequenced data set, temporary storage queue name, a transient data queue, or the name of a DataInterchange MQSeries Queue profile member. If this field indicates how the data is processed, it can specify the name of a response program that gains control after the transaction is translated, or the name of a response transaction that gains control after the transaction is translated. You must also enter an application file type.

You can use any combination of characters in the ALPHANUM table, up to 8 characters. Embedded blanks are not allowed. See “ALPHANUM and CHARSET Tables” on page 8-2 for more information.

<p>In this field:</p> <p>Application file type</p>	<p>Enter:</p> <p>This field defines the file type of the <i>Application file name</i>.</p> <p>For MVS, this field is either left blank if the application file name field is a ddname, or the value of MQ is supplied if the application file name field specifies a DataInterchange MQSeries Queue profile member.</p> <p>For CICS, valid values are:</p> <table> <tr> <th>Value</th><th>Description</th></tr> <tr> <td>MQ</td><td>DataInterchange MQSeries Queue profile member</td></tr> <tr> <td>TD</td><td>Transient data queue</td></tr> <tr> <td>TM</td><td>Temporary storage queue (main)</td></tr> <tr> <td>TS</td><td>Temporary storage queue (auxiliary)</td></tr> <tr> <td>VS</td><td>VSAM entry sequenced data set ddname</td></tr> <tr> <td>PG</td><td>Response program that is linked to after processing the application data</td></tr> <tr> <td>TX</td><td>Response transaction that is a CICS transaction started after processing the application data</td></tr> </table> <p>This field is used for transactions you receive, but not for transactions you send. The fault value is TS.</p>	Value	Description	MQ	DataInterchange MQSeries Queue profile member	TD	Transient data queue	TM	Temporary storage queue (main)	TS	Temporary storage queue (auxiliary)	VS	VSAM entry sequenced data set ddname	PG	Response program that is linked to after processing the application data	TX	Response transaction that is a CICS transaction started after processing the application data
Value	Description																
MQ	DataInterchange MQSeries Queue profile member																
TD	Transient data queue																
TM	Temporary storage queue (main)																
TS	Temporary storage queue (auxiliary)																
VS	VSAM entry sequenced data set ddname																
PG	Response program that is linked to after processing the application data																
TX	Response transaction that is a CICS transaction started after processing the application data																
<p>Record ID position</p>	<p>For more information about the data processed by your response program or response transaction, see the <i>DataInterchange Programmer's Reference</i>.</p> <p>The position where the record ID begins. You provide the record ID values when you define the data structures for this data format. This field is not used for C and D records.</p> <p>This field value is required for raw data record formats. The sample purchase order has the record ID starting in the ninth character position.</p> <p>This field must be a numeric value.</p>																
<p>Record ID length</p>	<p>The length of the record ID value. This field is not used for C and D records.</p> <p>This field value is required for raw data record formats. The value must be a number from 1 to 16.</p>																
<p>Record ID data type</p>	<p>The data type of the record ID, such as character (CH). See Table 7-2 on page 7-17 for a list of the data type codes.</p> <p>This field value is required for raw data record formats.</p>																
<p>Beginning structure name</p>	<p>The name of the structure in this data format that signals the start of a transaction. The beginning structure cannot be a repeating structure.</p> <p>If data described by this format is going to be translated using the raw data format, then either a beginning or ending structure name must be provided. If neither is provided and a raw data translation is attempted, the translation will fail as DataInterchange will not be able to locate transactions within the application file.</p> <p>You can use any combination of characters in the ALPHANUM table, up to 16 characters. Embedded blanks are not allowed. See "ALPHANUM and CHARSET Tables" on page 8-2 for more information.</p>																

Defining Your Application Data

In this field: Ending structure name	Enter: The name of the structure in this data format that signals the end of a transaction. The ending structure cannot be a repeating structure.
 	<p>If data described by this format is going to be translated using the raw data format, then either a beginning or ending structure name must be provided. If neither is provided and a raw data translation is attempted, the translation will fail as DataInterchange will not be able to locate transactions within the application file.</p> <p>You can use any combination of characters in the ALPHANUM table, up to 16 characters. Embedded blanks are not allowed. See “ALPHANUM and CHARSET Tables” on page 8-2 for more information.</p>
 Trading partner ID field	<p>The name of the field in this data format that contains the internal trading partner ID, which is usually a customer, vendor, or DUNS number. This field value is required for raw data record formats.</p>
 	<p>If data described by this format is going to be translated using the raw data format, then either a field name must be provided that contains the internal trading partner ID value or the TPID keyword must be used on the PERFORM command. The TPID keyword provides a default internal trading partner ID value if either no field was identified to contain this value or if the field identified contains all blanks.</p> <p>The sample purchase order uses a DUNS number which relates to the vendor company name. For information about using a DUNS number, see the <i>Using Information Exchange Administration Services</i> manual.</p> <p>You can use any combination of characters in the ALPHANUM table, up to 16 characters. Embedded blanks and lower case characters are not allowed. See “ALPHANUM and CHARSET Tables” on page 8-2 for more information.</p>
 	<p>During receive translation, DataInterchange automatically uses the internal trading partner ID literal from the trading partner receive usage for this field. It overrides any other mappings to the field.</p> <p>During send translation, this value and the application data format ID form the key used to find the trading partner send usage.</p>

<p>In this field: Generic routing code</p>	<p>Enter: The name of the field in this data format that contains the generic routing code used to select the appropriate generic send usage. Generic routing code is an optional, 3-character code provided by the application to select the correct generic usage when no specific usage has been defined for the trading partner. This field is optional and is used only for raw data formats.</p> <p>You can use any combination of characters in the ALPHANUM table, up to 16 characters. See “ALPHANUM and CHARSET Tables” on page 8-2 for more information.</p>
	<p>During send translation, if a usage is not found using the internal trading partner ID, a translation table with the same name as the Data Format ID is used to select the internal trading partner ID and trading partner nickname used in enveloping. The trading partner ID field is used as the local value in the translation table for the trading partner selection. DataInterchange prefixes the routing code with an ampersand (&) and uses this value as the internal trading partner ID to find a generic usage. If none are found, DataInterchange uses an ampersand (&) with all blanks to find a default generic send usage.</p>
<p>Interchange sender fields</p>	<p>Begin adding a data format by typing the information requested on this panel. Press Enter to save the information. On the next panel, you will describe the structures and fields in this data format.</p> <p>If the data records contain a record ID which the translator uses to associate the record with its structure, you must enter the following fields. If, on the other hand, you include the structure names in the data records, or if you pass the data in one record, omit these fields.</p> <ul style="list-style-type: none"> Record ID position Record ID length Record ID data type Beginning or ending structure name Trading partner ID field Generic routing code field
<p>Interchange receiver flds</p>	<p>Begin adding a data format by typing the information requested on this panel. Press Enter to save the information. On the next panel, you will describe the structures and fields in this data format.</p> <p>If the data records contain a record ID which the translator uses to associate the record with its structure, you must enter the following fields. If, on the other hand, you include the structure names in the data records, or if you pass the data in one record, omit these fields.</p> <ul style="list-style-type: none"> Record ID position Record ID length Record ID data type Beginning or ending structure name Trading partner ID field Generic routing code field

Defining Your Application Data

In this field:

Application TP ID field

Enter:

If a field in this data format contains the application trading partner name, enter the name of the data format field here.

An Application Trading Partner represents an entity like a division within a company doing centralized EDI, or a hub customer dealing with a group of spoke trading partners; in other words, an internal trading partner.

EDI TP field

If a field in this data format contains the EDI trading partner name, enter the name of the data format field here.

An EDI Trading Partner is the traditional DataInterchange trading partner; the spoke trading partner; the trading partner external to the installation.

The Add Data Structure panel (TD07) is displayed. Use this panel repeatedly to define each structure in the data format. The data format is identified in the **Data format ID** field.

Block Copy Delete Insert Move				
TD07		Add Data Structure		1 to 17 of 20
Data format ID . . .	POSAMPLE			
Structure name . . .	POBASE			
Maximum use count . .	1			
Record ID				
Action	Field/Structure Name	Data Type	Length	Passed Separately?
-	SHIPTO	st		y
-	DETAIL	st		y
-	TOTAL	st		y

4. Complete the fields as shown, and press Enter.

In this field:

Structure name

Enter:

A name for referring to this structure. For the sample purchase order, the base structure name POBASE represents the entire structure of a purchase order.

You can use any combination of characters in the ALPHANUM table, up to 16 characters. Embedded blanks are not allowed. See "ALPHANUM and CHARSET Tables" on page 8-2 for more information.

In this field:

Maximum use count

Enter:

The maximum number of times this structure may occur in the application data. If the structure occurs once, enter 1. You can enter any number from 1 through 32766 to specify a finite amount of structures. To specify an infinite amount of structures, enter 32767. The maximum use count determines:

- The number of occurrences the translator processes if the structure is not passed separately, or if a structure is passed separately, as long as at least one structure is nested within a structure other than the base structure
- The number of times the Interactive Entry Facility presents the data entry panel for this structure

If a structure is passed separately, and all structures are a part of the base structure, the translator does not use the value in this field, but the Interactive Entry Facility (IEF) uses this field to determine the number of occurrences to display or print.

If the structure is not passed separately (that is, physically part of the parent), and you are defining an ADF to create a FIXED standard, a value of 32767 (undefined) is not allowed during cReate standard. Additionally, if the structure has no substructures (that is, would not create a loop in the FIXED standard), then any maximum use count greater than 9998 is rejected by the cReate standard.

If you want DataInterchange to check the maximum use count of structures during translation to application, at least one structure must be nested a minimum of two levels deep. For example, your ADF has a base structure of POBASE with four nested structures: PURPOSE, NAME, LINES, and SUMMARY. If you want the maximum use count for the NAMES structure to be checked and enforced, delete the NAMES structure from the base structure and nest it in another structure, such as PURPOSE.

Record ID

A value that the translator looks for in the data record to associate it with this structure. This field value is used for raw data format. The record ID is case sensitive. You do not have to reformat the application data to include a structure name if you enter a value in the following fields on the Add, Copy, or Update Data Format panel:

- Record ID position
- Record ID length
- Record ID data type
- Beginning or ending structure name
- Trading partner ID field

Note: If the data type for the record ID is hexadecimal (HX), the record ID can contain only A through F and 0 through 9. If the data type is numeric, the record ID can contain only 0 through 9 and an optional sign. The translator converts the record ID to the appropriate data type before comparing it to the record ID field in the data.

Defining Your Application Data

In this field:

Action

Enter:

Select one or more actions shown on the action bar by typing the uppercase letter:

Block	Mark the boundaries of lines you want to move or copy
Copy	Copy a marked block below the line on which you enter the Copy action
Delete	Delete a line from the structure
Insert	Insert a blank line below the line on which you enter the Insert action
Move	Move a marked block below the line on which you enter the Move action, and erase the block from the old location

Field/Structure Name

The name of a field or the name of a structure nested in this structure. Each field and structure name must be unique within the application data format.

You can use any combination of characters in the ALPHANUM table, up to 16 characters. Embedded blanks are not allowed.

Data type

A code that indicates the type of data contained in the field. For example, the data type code for a structure is ST. See Table 7-2 on page 7-17 for a complete list of valid data types.

Length

The number of characters a data field can contain. This field is not used for data types that are structures. Special rules apply to the following data types:

BN, Bn, IT, and In	The length of the value must be 2 or 4 characters, depending on your application data.
DT	The length of the value must be 5, 6, 7, or 8 characters, depending on the date format of your application data.
Nn	The length of the value includes the sign if it is present. For example, the value -123 has a length of 4 characters.
R and Rn	The length of the value includes the decimal point and sign if they are present. For example, the value -123.4 has a length of 6 characters.
TM	The length of the value must be 4 or 6 characters.

Passed separately?

Indicates whether this structure is passed to the translator as a separate record or as part of its outer parent structure. For receive transactions, indicates whether the structure is output as a single record or as part of its outer parent structure. This field is used for structures only (data type ST). Valid values are Y (yes) and N (no).

5. The Add Data Structure panel (TD07) is redisplayed for you to add the next level of information. For the DETAIL structure, complete the fields as shown in the following panel, and press Enter.

This time you need to specify the length of the fields that are not structures. Also, all fields within a structure must be passed together, so you do not need to specify if they are passed separately or not. However, the DETAILDESC field is a structure, and you do have to specify if it is passed separately or not. In this example, the DETAILDESC structure is passed with the DETAIL structure. All of the fields on this panel are described in the previous step.

Note: The following screen assumes that you are using raw data. If you are using C and D records, do not include the DL record ID or the DLPONUMBER ADF field. For more information about the raw data, or C and D record format, see “Record Formats Used by DataInterchange” on page 7-1.

Block Copy Delete Insert Move

TD07 Add Data Structure 1 to 17 of 20

Data format ID . . . : POSAMPLE
Structure name . . . : DETAIL
Maximum use count . . : 1000
Record ID : DL

Action	Field/Structure Name	Data Type	Length	Passed Separately?
-	dlponumber	ch	8	-
-	dlrecid	ch	2	-
-	dlitnumb	ch	10	-
-	dlqty	n0	9	-
-	dlunits	ch	1	-
-	dluprice	n2	9	-
-	dludisc	n2	9	-
-	dltext	n2	9	-
-	detaildesc	st		n

The Add Data Structure panel (TD02) is redisplayed for you to add the next structure. You can add the DETAILDESC structure the same way you add all structures after you defined it on the Add Data Structure panel (TD07).

- 6. Press F3 (Exit) after you have added all your structures. The Data Formats panel (TD01) is redisplayed.
- 7. Type **s** next to POSAMPLE to verify that you have entered the correct structure, and press Enter.

Add Copy Delete List Print Update View Where used Structures
cReate standard

TD01 Data Formats 1 to 2 of 2

A	Data format ID	Description
	GENINV	General use invoice - Send
s	POSAMPLE	Sample purchase order

The Data Structures Hierarchy panel (TD06) is displayed. This panel shows how each structure is related to the previous structure. It is important that this hierarchy is accurate because DataInterchange uses this information during translation.

Defining Your Application Data

Add Copy Delete Update View		
TD01	Data Formats	1 to 2 of 2
A Data format ID GENINV POSAMPLE		
TD06	Data Structures Hierarchy	1 to 5 of 5
Data format ID : POSAMPLE Description : Sample purchase order		
Action	Structure	
—	POBASE	
—	SHIPTO	
—	DETAIL	
—	DETAILDESC	
—	TOTAL	

- When you have reviewed the Data Structures Hierarchy panel (TD06), press F12 (Cancel) to return the Data Formats panel (TD01).
- On the Data Formats panel (TD01), type **P** in the action column next to the sample purchase order. A report similar to Figure 7-6 is generated for you to verify the definition you just entered.

1APP1	POSAMPLE	Application Data Format	Date: 95/12/08 Time: 10:16:21
Description	Sample purchase order		
Base structure name	POBASE		
Application file name . . .			
Application file type . . .			
Record ID position	9		
Record ID length	2		
Record ID data type	CH		
Beginning structure name . .	SHIPTO		
Ending structure name . . .			
Trading partner ID field . .	SHDUNS		
Generic routing code field .			
Field/Structure Name	Type	Length/Max Use	Passed Separately?
POBASE	ST	1	Y
SHIPTO	ST	1	Y
SHPONUMBER	CH	8	
SHRECID	CH	2	
SHNAME	CH	30	
SHSTREET	CH	30	
SHCITY	CH	15	
SHSTATE	CH	2	
SHZIP	N0	9	
SHPHONE	N0	10	
DETAIL	ST	1000	Y
DLPONUMBER	CH	8	
DLRECID	CH	2	
DLITMNUMB	CH	10	
DLQTY	N0	9	
DLUNITS	CH	1	
DLUPRICE	N2	9	
DLUDISC	N2	9	
DLEXT	N2	9	
DETAILDESC	ST	3	N
DDPONUMBER	CH	8	
DDRECID	CH	2	
DDDESC	CH	30	
TOTAL	ST	1	Y
TLPONUMBER	CH	8	
TLRECID	CH	2	
TLAMOUNT	N2	9	

Figure 7-6. Data Format for the Sample Purchase Order

- Press F3 (Exit) to return to the Administrator's Menu (MP01).

Data Types for Application Data Formats

Table 7-2 lists the valid data types for ADF fields and structures. It also includes the valid mapping data type associated with the ADF data type. The mapping column applies to EDI standard mapping and does not apply to Fixed-to-Fixed mapping.

Most ADF data types use the same format for storing, displaying, and printing data. However, sometimes the storage format is different from the format used to display or print data. These differences are noted in the table.

Table 7-2 (Page 1 of 4). Data Types Used in Data Formats

Data Type	Mapping	Description
A	A, AN, ID	Alphabetic Any combination of characters from the ALPHANUM table, except the digits 0 through 9.
AC	A, AN, ID, Nn, Rn, DT, TM	Application control A field that contains a control number by which the application identifies the transaction. A purchase order number is an example. The data itself is alphanumeric. A data format can contain only one field of this type. During transaction mapping, you can specify the application control as a concatenation of up to eight fields. The concatenated application control overrides the AC data type. This data type does not apply to record IDs. An AC data type is assumed to be the same as AN during value validation at translate time.
AN	A, AN, ID, Nn, Rn, DT, TM	Alphanumeric You can use any combination of characters in the ALPHANUM table, up to the length of the field.
Bn	AN, ID, Nn, Rn, DT, TM	Binary (unsigned) Storage format: Data with a binary format with <i>n</i> implied decimal places. A value of 2.3 defined as a 2 byte B2 field would be stored as 1110 0110 (X'E6' or decimal 230). This is the same format as IT or In data, but binary data is not signed and therefore all values are considered positive.
BN	AN, ID, Nn, Rn, DT, TM	Binary (unsigned) Any combination of 0 through 9 without a sign (+ or -). Storage format: The binary equivalent of a numeric value in either 2 or 4 bytes, depending on the length of the field. Example: The value 23 is stored as 0000 0000 0001 0111 (X'0017').
CH	A, AN, ID, Nn, Rn, DT, TM	Character Any combination of characters up to the length of the field.

Defining Your Application Data

Table 7-2 (Page 2 of 4). Data Types Used in Data Formats

Data Type	Mapping	Description
DT	DT	<p>Date</p> <p>A string of 5 to 8 digits, depending on the date format that is used. The acceptable date formats are:</p> <ul style="list-style-type: none"> <i>ddmmyy, ddmmyyyy, ddyymm, ddyyyyym, dddy, dddyyy</i> <i>mmddy, mmddyyy, mmyy, mmyyyd</i> <i>yymm, yyyymm, yydd, yyddmm, yydd, yydd</i> <p>This data type does not apply to record IDs. When a DT ADF field is mapped to a DT element, data is treated as character data.</p>
FN		<p>File name</p> <p>A field that contains the name of a file whose entire contents are mapped to a binary segment. When not mapped to a binary segment, a field with data type FN is treated as if the data type were AN. See Appendix B, "Using the 841 Transaction Set" for more information.</p>
Hn	AN, ID, Nn, Rn, DT, TM	<p>Hexadecimal</p> <p>Hexadecimal data with <i>n</i> implied decimal places.</p> <p>This format is treated just like a Bn field when it is being mapped to a numeric data element, and is treated just like an HX field when it is being mapped to an alpha data element.</p>
HX	AN, ID, Nn, Rn, DT, TM	<p>Hexadecimal</p> <p>Any combination of 0 through 9 and A through F up to twice the length of the field.</p> <p>Storage format: Hexadecimal, where the length of the field determines the number of bytes used to hold the value.</p>
ID		<p>Identifier</p> <p>The ID data type is equivalent to the AN data type.</p>
In	AN, ID, Nn, Rn, DT, TM	<p>Integer (signed)</p> <p>Storage format: Data with a binary format with <i>n</i> implied decimal places. A value of 2.3 defined as a 4-byte I2 field would be stored as 0000 0000 1110 0110 (X'E6' or decimal 230).</p>
IT	AN, ID, Nn, Rn, DT, TM	<p>Integer (signed)</p> <p>Storage format: The binary equivalent for a positive number or the two's complement binary equivalent for a negative number, in 2 or 4 bytes, depending on the length of the field.</p> <p>Example: The value +23 is stored as 0000 0000 0001 0111 (X'0017'). The value -23 is stored as 1111 1111 1110 1001 (X'FE9').</p>
Ln	AN, ID, Nn, Rn, DT, TM	<p>Decimal (leading sign)</p> <p>Zoned decimal data with <i>n</i> implied decimal places and a leading sign.</p>
N	AN, ID, Nn, Rn, DT, TM	<p>Numeric</p> <p>Any combination of 0 through 9 and an optional sign (+ or -). The length includes the sign.</p> <p>When mapping data elements defined as data type N in EDIFACT standards, use data type R.</p>

Table 7-2 (Page 3 of 4). Data Types Used in Data Formats

Data Type	Mapping	Description
Nn	AN, ID, Nn, Rn, DT, TM	<p>Numeric</p> <p>Any combination of 0 through 9, an implied decimal point with <i>n</i> places to the right of the decimal, and an optional sign (+ or -). Using N alone is the same as using N0 (N zero). The length includes the sign.</p> <p>Example: N2 for a value of 23949 is interpreted as 239.49.</p>
PD	AN, ID, Nn, Rn, DT, TM	<p>Packed decimal</p> <p>Any combination of 0 through 9 with a sign (+ or -). The length defines the number of bytes used to hold the value in external format (minus the sign position).</p> <p>Storage format: The packed decimal equivalent, followed by the sign in the low-order 4 bits of the last byte. The sign is either 1111, 1100, or 1010 for a positive value; or, 1101 or 1011 for a negative value.</p> <p>Example: The value +123 is stored as 0001 0010 0011 1111 (X'123F'). The value -123 is stored as 0001 0010 0011 1101 (X'123D').</p>
Pn	AN, ID, Nn, Rn, DT, TM	<p>Packed decimal</p> <p>Packed decimal data with <i>n</i> implied decimal places.</p>
R	AN, ID, Nn, Rn	<p>Real</p> <p>Numeric data that requires a decimal point for fractional values. The decimal point is optional for integers. A sign (+ or -) is optional for positive numbers. Positive is assumed if a sign is not present. The length includes the decimal point and sign if they are present.</p> <p>You should use this data when mapping data elements defined as data type N in EDIFACT standards.</p> <p>Examples: 23.949, +23.949, -23949, -39846.7</p>
Rn	AN, ID, Nn, Rn	<p>Real</p> <p>Signed or unsigned numeric data with a minimum of <i>n</i> significant decimal places. The length includes the decimal point and sign. Any combination of 0 through 9 with a sign (+ or -).</p>
ST		<p>Structure</p> <p>A grouping of fields, such as all the fields making up the heading for an invoice or the fields making up one line item of an invoice. Length does not apply to structures.</p>
TM		<p>Time</p> <p>A string of four digits in the form <i>hhmm</i> or six digits in the form <i>hhmmss</i>. The time format uses a 24-hour clock, where the hour is specified as 00 to 23 for X12 and 00 to 24 for EDIFACT.</p> <p>This data type does not apply to record IDs.</p>

Defining Your Application Data

Table 7-2 (Page 4 of 4). Data Types Used in Data Formats

Data Type	Mapping	Description
ZD	AN, ID, Nn, Rn, DT, TM	<p>Zoned decimal</p> <p>Any combination of 0 through 9 with a sign (+ or -). The length defines the number of characters used to represent the value in the external format. The external length requires an extra position for the sign.</p> <p>Storage format: The zoned decimal equivalent in the low-order 4 bits of a byte and 1111 in the high-order 4 bits. The sign appears in the high-order 4 bits of the low-order byte and is either 1100 for a positive value or 1101 for a negative value. The length of the field determines the number of bytes used to store the value.</p> <p>Example: The value +123 is stored as 1111 0001 1111 0010 1100 0011 (X'F1F2F3'). The value -123 is stored as 1111 0001 1111 0010 1101 0011 (X'F1F2D3').</p>
Zn	AN, ID, Nn, Rn, DT, TM	<p>Zoned decimal</p> <p>Zoned decimal data with <i>n</i> implied decimal places and a trailing sign. Any combination of 0 through 9 with a sign (+ or -).</p>

Reformatting Data into Control (C) and Data (D) Records

When you use control (C) and data (D) records, each transaction has one C record. The number of D records varies as follows:

- One D record for each transaction—no data structures are passed separately
- One D record for each data structure—all data structures are passed separately
- D records with variable numbers of data structures—some passed separately, some together

An example of each variation follows. For a description of the C record, see the *DataInterchange Programmer's Reference*.

One D Record for Each Transaction

The following is an example of reformatted data when all structures are passed together:

```
Columns
      1      2      3      4      5      6      7      8      9
123456789012345678901234567890123456789012345678901234567890123456789
-----
CSPSTT16  AFTSU09  NY
DP0123456  092091C321          Our store will be closed for vacationfrom July 3 to July 14.    ...
Z
CSPSTT16  AFTSU09  NY
```

Figure 7-7. Reformatted Data—All Structures Passed Together

The following chart shows the data format for this variation:

Structure	Max use	Passed separately?
POBASE		
POHDR	1	N
PONOTE	3	N
POLINEITEM	10	N
PODESCR	2	N
POTOT	1	N

Notes:

- 1. The letter D in the first position identifies the data record.
- 2. Structure names do not appear in the data.
- 3. All occurrences defined in the data format must be present in the application file, even if there is no data for them. The third occurrence of PONOTES, for example, is 30 blank characters. Similarly, the application data must account for ten line items, even though eight are blank.

One D Record for Each Data Structure

The following is an example of reformatted data when all structures are passed separately.

Columns

1	2	3	4
---	---	---	---

12345678901234567890123456789012345678901234567

CSPSTT16 AFTSU09 NY

DPOHDR P0123456 092091C321

DPONOTE Our store will be closed for v

DPONOTE acation from July 3 to July 14

DPOLINITEM 000002EA00006995

DPODESC Glass fly rod

DPODESC 7 ft. #6 line

DPOLINEITEM 000002EA00004995

DPODESC Single action

DPODESC fly reel

DPOTOT 000023980

Z

CSPSTT16 AFTSU09 NY

Figure 7-8. Reformatted Data—All Structures Passed Separately

The following chart shows the format for this variation:

Structure	Max use	Passed separately?
POBASE		
POHDR	1	Y
PONOTE	3	Y
POLINEITEM	10	Y
PODESC	2	Y
POTOT	1	Y

Defining Your Application Data

Notes:

1. The letter D in the first position identifies each data record.
2. A structure name appears in positions 2 through 17.
3. The application data appears in positions 18 through *n*.
4. Because each record contains a structure name and is passed separately, the application file does not have to contain blank records to account for unused occurrences, as in the previous example. This is ideal for structures that occur a variable number of times.

Variable D Records

The following is an example of reformatted data when some structures are passed separately.

```
Columns
      1      2      3      4      5      6
1234567890123456789012345678901234567890123456789012
-----
CSPSTT16  AFTSU09  NY
DPOHDR      P0123456  092091C321
DPONOTE      Our store will be closed for v
DPONOTE      acation from July 3 to July 14
DPOLINEITEM  000002EA000006995Glass fly rod  7 ft. #6 line
DPOLINEITEM  000002EA000004995Single action  fly reel
DPOTOT      000023980
Z
CSPSTT16  AFTSU09  NY
```

Figure 7-9. Reformatted Data—Some Structures Passed Separately

The following chart shows the data format for this variation:

Structure	Max use	Passed separately?
POBASE		
POHDR	1	Y
PONOTE	3	Y
POLINEITEM	10	Y
PODESCR	2	N
POTOT	1	Y

Notes:

1. The letter D in the first position identifies each data record.
2. A structure name appears in positions 2 through 17.
3. The application data appears in positions 18 through *n*.
4. Because PODESCR is passed with POLINEITEM, the application file must always account for two occurrences of PODESCR.

Changing Your Application Data Format after Mapping

When you change a data format that is already mapped, you must also do one or more of the following:

- Map any new fields you want to appear in the mapping, and regenerate the translation map (control string).
- Delete or modify the segment and data element mappings that refer to structures or fields deleted from the data format, then regenerate the translation map (control string).
- Regenerate the translation map (control string).

Creating an Application Data Format for an Empty Transaction

If you have a transaction that has no application data and is created only by literal or accumulator values, you can create an empty application data format for that transaction.

If you are using C and D records, create an ADF with only a base structure. Ignore any messages about the format containing incomplete structures. To send the transaction, the input file should have the normal C record along with a single D record containing no data.

If you are using raw data formats, create an ADF with a base structure that contains a record ID field and an internal trading partner ID field. To send the transaction, the input file should have one record containing the record ID and the internal trading partner ID.

Translating Application Data to Application Data (Fixed-to-Fixed)

Fixed-to-Fixed refers to translation from one application data format to another. The Fixed-to-Fixed function is available only for outbound processing.

During Fixed-to-Fixed translation, there is an application file that contains the source data. This file has the same considerations as any other source file that DataInterchange translates. The source file can either contain C and D records (and, thus, have multiple application data formats), or it can contain raw data (and, thus, have a single format defined by the RAWFMTID keyword). The target translated data may also be written to the file in a C and D record format or in a raw data format.

To use raw data as your source file, the file must have:

- A fixed location in every record that contains a unique code that identifies that record
- A nonrepeating structure that identifies the start, the end, or both the start and end of a transaction (logical record)
- A field that contains the internal trading partner ID, which can be identified as such. You may assign a constant trading partner ID in the translate command itself for those instances where a field in the data does not exist (for example, keyword TPID added to performs).

Although the source application data format must meet either raw data or C and D specifications, it would not be necessary for the target application data format to have raw data specifications, for raw data to be output. Because the target data will not be used as input to a translation, it is not required to follow the rules for raw data. Raw data output means the records will be written to the output file exactly as described by the application data format definition.

Setting up Fixed-to-Fixed Translation

Fixed-to-Fixed translation requires a map to direct the movement of data between application data formats. You create the map by using the mapping facility that displays a standard transaction and allows structures and fields from the application data format to be associated with segments and data elements in the standard transaction. When using Fixed-to-Fixed translation, the target application data format must be defined as a standard so that a map can be created. However, you do not need to create this standard. You define the target application data format just as the source was defined and then use the new `cReate standard` facility command (action item), to convert the application data format definition into a standard. This conversion only needs to be done a single time once the target ADF is defined.

The steps for implementing Fixed-to-Fixed translation are:

1. Define the source application data format.
2. Define the target application data format.
3. Use the `cReate standard` facility command to convert the target application data format defined in step 2 into a standard.

The `cReate standard` command generates a standard format that can be viewed with DataInterchange option 5 from the Administrator's Menu. The rules for `cReate standard` are:

- The standard ID would be the application file name.
- The standard description would be the application data format ID.
- The envelope type of the standard would be F to indicate this is a FIXED format standard.
- The transaction ID would be a fixed value of FFFFFFFF.
- The transaction description would be the application data format description.
- Each structure within the application data format would become a segment within the standard. The segment name would be assigned by the create command (SEG1, SEG2, SEG3, and so on). If all the structure names are 8 characters or less in length, the structure name will be used as the segment name rather than the arbitrary names of SEG1, SEG2, and so on.
- The description for the segment would be the structure name from the application definition.
- A structure that repeats but does not contain embedded structures would become a repeating segment.
- A structure that contains embedded structures becomes a loop.

Note: The maximum nesting in the application data definition would be 15 because only 15 levels of looping are allowed in a standard.

- Each field within a structure would become a data element within the standard. The data element name would be assigned by the create command. (DE1, DE2, and so on). If all the field names are 8 characters or less in length, the field name will be used as the data element name rather than the arbitrary names of DE1, DE2, and so on.
- The description for the data element would be the field name from the application definition.
- The minimum and maximum data element lengths would be equal to the field length, and the data elements data type would be equal to the field data type.
- Mandatory segment or data element checks are not performed during Fixed-to-Fixed translation. Therefore, making the segments or elements mandatory will have no affect if data is not passed to certain fields, segments, and elements. Mandatory segment or element errors will NOT be issued if data is not passed to the defined segments and elements.
- All data elements are created with REQ DES of M.

4. Create the map between the source application data and the target application data (standard).
5. Create a trading partner usage(s) for this map by using the Where used option. The usages define which trading partners are going to use this map.
6. Generate the control string for the map.

7. Perform translations by issuing commands:

```

PERFORM TRANSLATE AND ENVELOPE ...
      or
PERFORM TRANSLATE TO STANDARD ...
PERFORM ENVELOPE ...
PERFORM TRANSLATE AND SEND ...
PERFORM ENVELOPE AND SEND ...

```

The **PERFORM TRANSLATE AND ENVELOPE** command will perform Fixed-to-Fixed translation if the standard has an envelope type of 'F'. Otherwise, standard translation is done. **PERFORM TRANSLATE TO STANDARD** and **PERFORM ENVELOPE** commands can be issued separately for Fixed-to-Fixed translation. The **PERFORM TRANSLATE TO STANDARD** will create a fixed file without an interchange wrapped around it unless an envelope override is specified on the trading partner usage.

Regardless of whether commands are issued separately or together, the resulting output will be placed in the file identified by the **FIXEDFILEID** keyword. If the **FIXEDFILEID** keyword is not used, the resulting output will be placed in the ddname formed by the concatenation of the standard ID (which was the Application file name in the application data format that created the standard) with the file suffix field in the trading partner profile.

DataInterchange places copies of data going to multiple trading partners in different files. To make the ddnames unique, a 2-character file suffix from the trading partner profile is appended to the application file name. You must update all trading partner profiles that use this map with the associated file suffix. To write all resulting data to the same file, use the **FIXEDFILEID** keyword.

An example of a command for Fixed-to-Fixed translation is:

```

PERFORM TRANSLATE AND ENVELOPE
      WHERE APPFILE(DISOURCE) RAWFMTID(DIUSERFROMADF) RAWTEST(Y)
      FIXEDFILEID(DITARGET) RAWDATA(Y)

```

8. To send the data to your trading partner, issue commands:

```

PERFORM SENDFILE ...
PERFORM SEND ...

```

| **Note:** These commands are valid only if you are using the IBM Global Network.

The sequence of actions for this translation is shown in Figure 7-10 on page 7-26.

Defining Your Application Data

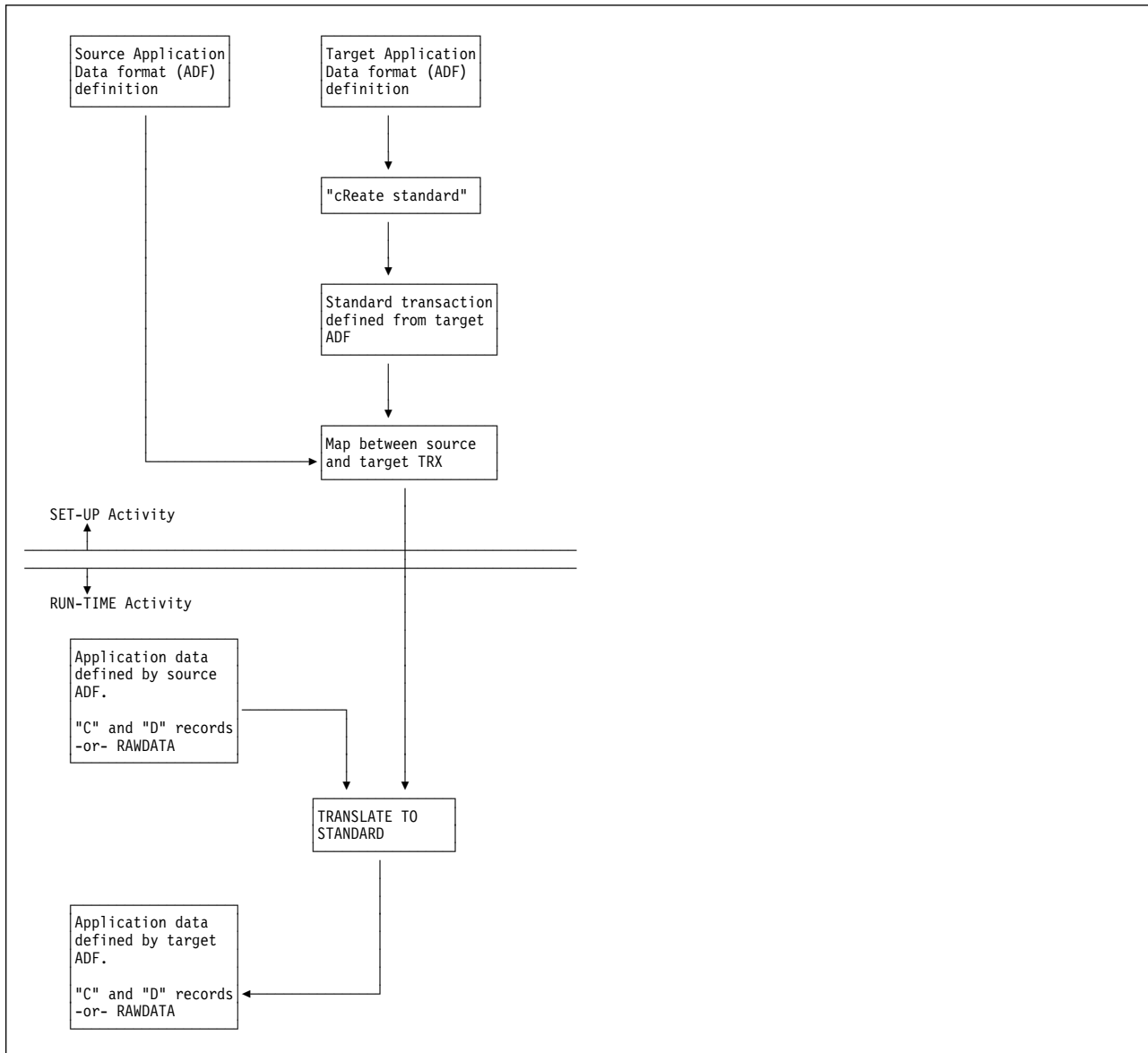


Figure 7-10. Fixed-to-Fixed

FIXED Standards without an Associated ADF

It is also possible to have a Fixed-to-Fixed translation that involves a source ADF and a fixed standard that is not associated with an ADF (for example, not cReated from an ADF). A FIXED standard is simply a standard that has an envelope type of F. The F indicates that only fixed length segments and data elements are present, therefore delimiters are not used.

To get the desired results, all control strings that use that standard (which are listed when you attempt to make the change) must be newly generated. Results are not guaranteed if the envelope type for a standard is changed to F, but the control strings are not regenerated.

Blank Application Data Format Worksheet

You may want to copy the following worksheet to prepare your application data format.

Table 7-3. Blank Worksheet for Your Application Data Format

[illegible]

Chapter 8. Translation and Validation Tables

Translation and Validation Tables - Usage and Tips	8-1
Tables Supplied by DataInterchange	8-2
ALPHANUM and CHARSET Tables	8-2
Alphanumeric Table (ALPHANUM)	8-3
Character Set Table (CHARSET)	8-3
File Name Table (FILENAME)	8-3
Monocase Table (MONOCASE)	8-3
Program Name Table (PRGMNAME)	8-3
Translation Tables	8-4
When to Use Type T Translation Tables	8-4
When to Use Type R (Reverse) Translation Tables	8-5
Adding a Translation Table	8-5
Updating a Translation Table	8-8
Validation Tables	8-9
Adding a Validation Table	8-10
Updating a Validation Table	8-12

Chapter 8. Translation and Validation Tables

When DataInterchange sends a transaction to one of your trading partners, it first translates the transaction from your application format to the EDI standard format that you and your partner have agreed to use. When DataInterchange receives a transaction from your trading partner, it translates the transaction from the EDI standard format to your application format before delivering it to your application. As part of the translation, DataInterchange can substitute one value for another. This way, the data is translated from your local format to the format required by a standard or trading partner for send transactions, and back to the form you prefer for receive transactions. DataInterchange can also verify that a field or data element contains one of the values from a list of acceptable values. To perform the substitution and verification, DataInterchange uses translation and validation tables.

A *translation table* contains data to translate differing values between your application and a standard or between your application and a trading partner's application. During send translation, the data in an application field is replaced by a corresponding value specified in the table. During a receive translation, the data in a data element is replaced by a corresponding value specified in the table. If the data in the application field does not match a table entry, a translation error occurs.

A *validation table* provides a list of acceptable values for a data element. During send translation, the application field data is verified against the table. During receive translation, the data element is verified against the table. If the data in the application field does not match a table entry, a translation error occurs.

In addition to the add and update commands described on the following pages, translation and validation tables can also be copied, deleted, printed, or viewed. The **LIST** command can also be used to find a table that is not on the current screen or current list.

Translation and Validation Tables - Usage and Tips

Use validation and translation tables to handle:

- Differences between your data and your trading partner's data; for example, different part numbers used for the same item.
- Conflicts between application data and EDI standards. For example, your application uses a code for a unit of measure that does not appear in the standard. You can either create a translation table to substitute a standard code, or you can update the validation table to include your code. If you change a standard validation table, however, it is no longer standard, which can cause problems if your trading partners do not use the same code.
- Differences among your trading partners. The validation tables supplied by DataInterchange apply to the standard itself and, therefore, to *all* trading partner transactions based on the standard. You can think of these tables as the defaults for all trading partners. You can override the use of these tables by providing the ID of another table when defining data elements for partner-specific transactions. You can also omit the table name when mapping, and no verification occurs for that data element.

All translation tables are user defined. Depending on the data you exchange and the agreements you have, you can define tables that apply to one trading partner or to several. However, these tables are always mapping-specific in that there are no default tables for an entire standard as there are for the validation tables.

Translation and Validation Tables

Some additional points to consider are:

- Using validation and translation tables can slow performance. However, because DataInterchange remembers the results of the most recently used validation or translation table entries, the decrease may not be significant, unless you are using many different validation and translation tables in one transaction.
- To save storage and reduce look-up time, you can delete entries from the default validation tables if the entries do not apply to your data.
- To save time when creating a new table, you can copy a table that is similar and update the copy.

If the translation or validation tables you are planning to define in DataInterchange are already maintained by an existing application or are changed by many different people, it may make sense to maintain this table outside DataInterchange and use a field exit to check or replace the value at translation time. For more information about invoking a field exit, see Chapter 9, “Mapping Your Application Data to an EDI Standard Transaction Set.” For more information about writing a field exit, see the *DataInterchange Programmer's Reference*.

Tables Supplied by DataInterchange

The following tables are loaded by DataInterchange during initialization to improve performance. In DataInterchange for CICS, a temporary storage queue named EDITV00 is created to hold the table information. In DataInterchange for MVS, the table information is loaded into virtual storage. If you modify any of the following tables, you must purge the EDITV00 TS queue in DataInterchange for CICS or sign off DataInterchange in DataInterchange for MVS before the changes can take effect. Do not delete these tables.

Table	Description
ALPHANUM	Alphanumeric validation table
CHARSET	Character set validation table
FILENAME	File name character validation table
MONOCASE	Uppercase character translation table
PRGMNAME	Program name character validation table

ALPHANUM and CHARSET Tables

DataInterchange provides default ALPHANUM and CHARSET tables. In addition, DataInterchange provides a means to specify multiple ALPHANUM and CHARSET tables. These tables are used by the DataInterchange translation process and administrative functions. You can add user-defined tables for administrative panel edits and translation processes.

Note: Default references are to the ALPHANUM and CHARSET tables, but you can also employ user-defined tables identified in the APPSDEF profile member.

The following four overrides specify the ALPHANUM and CHARSET tables:

- No specific override - Pre-loads tables named ALPHANUM and CHARSET which are shipped with the product. This is the default.
- Specification in the APPSDEF profile - Pre-loads tables named in the APPSDEF profile. If the fields are blank, the defaults are used. For batch translation processing, the APPLID parameter on the EDIFFUT program call is used to specify the APPSDEF to use. For realtime translation processing, Application ID is specified in the CONTRECV profile, and APPLID in the Utility control block. For Application Programming Interfaces (API), the application ID parameter is specified with the API

initialization call. The APPLID keyword may also be specified on outbound Utility PERFORM commands. A special set of characters is defined in the alphanumeric table.

- Specification in the trading partner usage - Reloads the tables specified. If the fields are blank, the default is used or reloaded. For performance considerations, this override has very limited use because of the reloading process.
- EDIFACTUNB01 specification - Extracts this information and attempts to reload the ALPHANUM table only. If the table is not found, the default is used.

Alphanumeric Table (ALPHANUM)

A special set of characters is defined in the alphanumeric table (ALPHANUM). These characters are a subset of the characters in the character set table. This table is used to validate entries restricted to alphanumerics. If you find that you must add characters to the alphanumeric character set, add them by updating the ALPHANUM table, or create an additional table.

Character Set Table (CHARSET)

The character set validation table (CHARSET) is used throughout DataInterchange to determine valid characters. Only the characters that appear in the table are valid. This table is used to check for valid characters during data entry and translation.

If you need to change this set of characters, you can update the table or create an additional table. For example, you may have to add characters to this table if you want to use accented characters that are not normally used in your language.

Note: If code page conversion is in effect for your terminal, the interpretation of the characters you enter may not be what you expect.

You can only update other tables with characters from the character set table. If you find that you cannot update another table with characters you want, you will have to first add them to the character set table.

File Name Table (FILENAME)

A special set of characters is defined in the file name table (FILENAME). These characters are a subset of the characters in the character set table. This table is used to validate the characters entered for ddnames on application data format (ADF) panels and trading partner transaction usage panels.

Monocase Table (MONOCASE)

The MONOCASE table is a translation table used to convert character input to uppercase. It contains pairs of values: the first column contains the uppercase value and the second column contains the characters to be changed to uppercase. Characters found in this table are monocased. If for some reason you need to change the monocasing characters, you can update the MONOCASE table.

Program Name Table (PRGMNAME)

A special set of characters is defined in the program name table (PRGMNAME). These characters are a subset of the characters in the character set table. This table is used to validate the characters entered for program names in trading partner transactions. Post-translation exits, pre-translation exits, and user exits are the fields where this table is used.

Translation Tables

With translation tables, you and your trading partners can use your own codes and values, enabling you to exchange information and meet EDI standard requirements. For example, if you and your supplier use different part numbers for the same items, you can create a translation table to change your part numbers to the supplier's part numbers before sending a purchase order, and change the supplier's part numbers to your part numbers when you receive the invoice. Your table may look like this:

Local Value	Standard or Trading Partner Value
GLF8088	FR0100
GLF8588	FR0600
GRF8788	FR0800

Translation tables contain paired values arranged in two columns. Column 1 has key source values, and each value can occur only once. Column 2 has translation data values, and values can occur more than once.

DataInterchange has two types of translation tables: type T and type R. In a type T translation table, column 1 contains the local application value, and column 2 contains the trading partner or standard value. In a type R translation table, column 1 contains the trading partner or standard value, and column 2 contains the local application value.

When transactions are sent, local values are translated to standard or trading partner values. When transactions are received, standard or trading partner values are translated to local values.

You can specify translation tables with a data element mapping on the Special Handling for Sending panel (TP30) and Special Handling for Receive panel (TP31). For more information, see "Special Handling" on page 9-59.

When to Use Type T Translation Tables

When each value of your application data translates to only one standard or trading partner value, use a type T translation table. For example, your application always specifies the month as a two-digit number, but your trading partner prefers the full name of the month. You can create a translation table with the following entries:

Local Value	Standard or Trading Partner Value
01	January
02	February
03	March
04	April
05	May
06	June
07	July
08	August
09	September
10	October
11	November
12	December

Because of the one-to-one relationship between the local value and the trading partner value, this type T table works equally well for both send and receive transactions.

When more than one of your application values translate to a single standard or trading partner value, use a type T translation table. For example, your application uses both D0Z and 12 for dozen, but the standard accepts only DZ. You can create a type T translation table with the following entries:

Local Value	Standard or Trading Partner Value
BOXES	BX
CASES	CS
D0Z	DZ
EACH	EA
12	DZ

Because of the many-to-one relationship between the local values and the standard values, this type T table works best for send transactions. It can be used for receive transactions, but DataInterchange stops on the first matching value in column 2. In this example, DZ is always changed to D0Z.

When to Use Type R (Reverse) Translation Tables

When more than one trading partner value translates to a single value in your application, use a type R translation table. For example, your trading partner's application dates items by month using a two-digit number, but you want to date the items by quarter. You can create a type R translation table with the following entries:

Standard or Trading Partner Value	Local Value
01	1st Qtr
02	1st Qtr
03	1st Qtr
04	2nd Qtr
05	2nd Qtr
06	2nd Qtr
07	3rd Qtr
08	3rd Qtr
09	3rd Qtr
10	4th Qtr
11	4th Qtr
12	4th Qtr

Because of the one-to-many relationship between the local values and the trading partner values, type R tables work best for receive transactions. They can be used for send transactions, but DataInterchange uses the first matching value in column 2. In this example, 1st Qtr is always changed to 01.

Adding a Translation Table

To add a translation table, follow these steps:

1. From the Administrator's Menu (MP01), select *Translation and Validation Tables*. The Translation and Validation Tables panel (TM01) is displayed.
2. Type **A** in the action column next to any item, and press Enter.

Translation and Validation Tables

Add	Copy	Delete	List	Print	Update	View
TM01			Translation and Validation Tables			1 to 3 of 3
Action	Table ID	Type	Description			
a	ALPHANUM	V	Alphanumeric validation table			
-	CHARSET	V	Character set validation table			
-	TRANS	T	Translation Table			

The Add Translation or Validation Table panel (TM02) is displayed.

TM01	Translation and Validation Tables	1 to 4 of 4	
Action	Table ID	Type	Description
a	ALPHANUM	V	Alphanumeric validation table
-			
-	TM02	Add Translation or Validation Table	
-	Table ID	units	
	Type	+	
	Description	Convert unit codes to standard	

3. Complete the fields as follows:

In this field:

Table ID

Type

Description

Enter:

A unique name to identify the name you want for this table.

The type of translation table you are creating. Valid values are:

Code Description

T Column 1 contains the local application value, and column 2 contains the trading partner or standard value.

R Column 1 contains the trading partner or standard value, and column 2 contains the local application value.

A brief optional description of the table you are creating.

4. Press Enter to save this information. The Add Translation Table panel (TM09) is displayed.

TM01	Translation and Validation Tables	1 to 4 of 4	
Action	Table ID	Type	Description
a	ALPHANUM	V	Alphanumeric validation table
-			
-	TM09	Add Translation Table	
-	Table ID	Type	Description
	UNITS	T	Convert unit codes to standard
	Local Value		Std or TP Value
	Length . .	05	Length . . 02
	Data Type.	ch	Data Type. ch

5. Complete the fields as follows:

In this field:	Enter:						
Length	The maximum length of entries for local values and standard or trading partner values. If the length is less than 10, use a zero in front of the single digit, for example, enter 02 for 2. The maximum length of the local value is 35 characters. The maximum length of the standard or trading partner value is 63 for a type T table and 35 for a type R table. The combined length of both columns cannot exceed 68 characters. For example, if the local value length is 20 for a type T table, the length of the standard or trading partner value is limited to 48. However, if the local value length is 20 for a type R table, the length of the standard or trading partner value is limited to 35, the maximum value.						
Data Type	The data type for entries in the column. Valid entries are: <table><tr><td>Code</td><td>Description</td></tr><tr><td>CH</td><td>Character data</td></tr><tr><td>R</td><td>Numeric data</td></tr></table>	Code	Description	CH	Character data	R	Numeric data
Code	Description						
CH	Character data						
R	Numeric data						

6. Press Enter to save this information. The Add Translation Table Entries panel (TM10) is displayed.

The width of the columns is determined by the values you entered on the previous panel. The order of the columns depends on whether the translation table is type T or R.

For a type T translation table, the panel looks like this:

TM10

Add Translation Table Entries

1 to 14 of 14

Table ID: UNITS

Description: Convert unit codes to standard

Local Value	Standard or Trading Partner Value
_____	_____
_____	_____
_____	_____
_____	_____
_____	_____
_____	_____

For a type R translation table, the panel looks like this:

TM10

Add Translation Table Entries

1 to 15 of 15

Table ID: UNITS

Description: Convert unit codes to standard

Standard or Trading Partner Value	Local Value
_____	_____
_____	_____
_____	_____
_____	_____
_____	_____
_____	_____

Translation and Validation Tables

7. In the *Local Value* column, enter a value used in your application. In the *Standard or Trading Partner Value* column, enter the value that corresponds to your application value. Repeat for each entry you want in this table.

Notes:

- a. Table entries are case sensitive: *abc* is not the same as *ABC*.
- b. The translation table also acts as a verification table. If DataInterchange encounters a code that is not in the table, it records an error. Therefore, every code used in the data should be in the translation table, even if the value does not change.

If you fill all of the available lines and press Enter, another panel of lines is displayed. If you do not need the extra lines, go to the next step.

8. Press Enter to save the table entries. The Translation and Validation Tables panel (TM01) is redisplayed.
9. Press F3 (Exit) to return to the Administrator's Menu (MP01).

Updating a Translation Table

To update a translation table, follow these steps:

1. From the Administrator's Menu (MP01), select *Translation and Validation Tables*. The Translation and Validation Tables panel (TM01) is displayed.
2. Type **U** in the action column next to the table you want to update, and press Enter.

Add	Copy	Delete	List	Print	Update	View
TM01			Translation and Validation Tables			1 to 4 of 4
Action	Table ID	Type	Description			
—	ALPHANUM	V	Alphanumeric validation table			
—	CHARSET	V	Character set validation table			
—	TRANS	T	Translation Table			
u	UNITS	T	Convert unit codes to standard			

The Update Translation Table Entries panel (TM13) is displayed. The order of the columns depends on whether the translation table is type T or R.

For a type T translation table, the panel looks like this:

Add	Delete	List
TM13		
Update Translation Table Entries		
1 to 5 of 5		
Table ID: UNITS Description Convert unit codes to standard_____		
Action	Local Value	Standard or Trading Partner Value
—	BOXES	BX
—	CASES	CA
—	DOZ	DZ
—	EACH	EA
—	12	DZ

For a type R translation table, the panel looks like this:

Add Delete List		
TM13		Update Translation Table Entries 1 to 5 of 5
Table ID: UNITS Description Convert unit codes to standard_____		
Action	Standard or Trading Partner Value	Local Value
-	BX	BOXES
-	CS	CASES
-	DZ	DOZ
-	EA	EACH

On this panel, you can change the description of the table and the table entries.

3. To delete entries, type **D** in the action column next to each entry you want to delete, and press Enter.
4. To display a specific entry at the top of the list, type **L** and the entry on the command line, and press Enter.
5. To add an entry, follow these steps:
 - a. Type **A** in the action column next to any item, and press Enter.
 - b. A blank line is inserted above that entry. On that line, enter the new entry values.

Note: Table entries are case sensitive: *abc* is not the same as *ABC*.

c. Press Enter.

6. When you are finished updating the table, press F3 (Exit) twice to return to the Administrator's Menu (MP01).

Note: Changes you make to a translation table affect the translation process as soon as the table is saved. This is different than changes to standards, ADFs, and maps, which do not affect translation until the control string is generated again.

Validation Tables

Some data elements must contain a specific value or one of a limited number of acceptable values. In this situation, you should create a validation table.

DataInterchange provides validation tables for most data elements in a standard with data type ID. For example, the X12 standard data element for unit-of-measure, 355, has a validation table that defines all the acceptable values for the data element, such as CA for case and EA for each. For new standards that are processed and shipped with DataInterchange, the naming convention for the validation tables is *eeeeesrrr*

where:

eeee

is a 1- to 4-character element number (for example, 355, 1001, 98)

s

is a 1-character standard ID. Valid values are:

Value	Description
E	EDIFACT
X	X12

Translation and Validation Tables

T EDIA (TDCC)
U UCS and WINS

rrr

is a 2- or 3-character release number. Year and release for EDIFACT (for example, 902 for EDIFACT 1990 Release 2). Version and release for TDCC, X12, UCS, and WINS (for example, 31 for X12 Version 3 Release 1).

For example, the validation table for element 355 for Version 3 Release 1 of X12 would be 355X31. UNTDI/TRADACOMS does not follow this convention. If the table does not exist under this name, the element needs to be viewed using the standards customization facility for the particular standard desired. For more information, see “Viewing Standards, Transaction Sets, Segments, Composite Elements, and Data Elements” on page 6-23.

These supplied validation tables are updated when you apply or import a standard. If you modify one of these tables, the Apply Standards program prints a report showing the table's contents before the standard was applied.

You can also create your own validation tables. For example, if you exchange transactions with a trading partner who has offices in three cities, you can create a validation table to ensure that a transaction sent to this partner has one of those three city names. Your validation table would look like this:

Entry	Description
Blairsville	Home office
Blue Ridge	South branch
Young Harris	North branch

You can specify validation tables with a data element mapping on the Special Handling for Sending panel (TP30) and Special Handling for Receiving panel (TP31). For more information, see “Special Handling” on page 9-59. However, the validation table is used only when the validation level is 1 or 2. If the validation level is zero, DataInterchange ignores the validation table. You specify the validation level on the Trading Partner Usage Overrides for Sending panel (TP26) or the Trading Partner Usage Overrides for Receiving panel (TP27). For more information, see Chapter 9, “Mapping Your Application Data to an EDI Standard Transaction Set.” If the validation level is 1 or 2, a translation error occurs if the value is not found in the table.

Adding a Validation Table

To add a validation table, follow these steps:

1. From the Administrator's Menu (MP01), select *Translation and Validation Tables*. The Translation and Validation Tables panel (TM01) is displayed, listing the existing tables.
2. Type **A** in the action column next to any existing table, and press Enter.

Add	Copy	Delete	List	Print	Update	View
TM01						Translation and Validation Tables 1 to 3 of 3
Action	Table ID	Type	Description			
a	ALPHANUM	V	Alphanumeric validation table			
—	CHARSET	V	Character set validation table			
—	TRANS	T	Translation Table			

The Add Translation or Validation Table panel (TM02) is displayed.

TM01 Translation and Validation Tables 1 to 3 of 3			
Action	Table ID	Type	Description
a	ALPHANUM	V	Alphanumeric validation table
<div> <div>TM02 Add Translation or Validation Table</div> <div> <div>Table ID</div> <div>xyzcity</div> </div> <div> <div>Type</div> <div>v</div> </div> <div> <div>Description</div> <div>Cities with XYZ offices</div> </div> </div>			

3. Complete the fields as follows:

In this field:

Table ID

Type

Description

Enter:

A unique name to identify the name you want for this table.

V for validation table.

A brief description of the table you are creating.

4. Press Enter to save this information. The Add Validation Table panel (TM03) is displayed.

TM01 Translation and Validation Tables 1 to 3 of 3			
Action	Table ID	Type	Description
a	ALPHANUM	V	Alphanumeric validation table
<div> <div>TM03 Add Validation Table</div> <div> <div>Table ID</div> <div>XYZCITY</div> </div> <div> <div>Type</div> <div>V</div> </div> <div> <div>Description</div> <div>Cities with XYZ offices</div> </div> <div> <div>Length of entries</div> <div>12</div> </div> <div> <div>Data type</div> <div>ch +</div> </div> <div> <div>Description of entries (Y/N)</div> <div>y +</div> </div> <div> <div>Length of descriptions . . .</div> <div>20</div> </div> </div>			

5. Complete the fields as follows:

In this field:

Length of entries

Data Type

Description of entries

Length of descriptions

Enter:

The maximum number of characters for an entry. The limit is 35 characters.

The data type for entries in the column. Valid entries are:

Code	Description
CH	Character data
R	Numeric data

Y if you want to include brief descriptions of the entries, or **N** if you do not.

The maximum number of characters for the descriptions. The limit is 63 characters. The combination of entry length and description length cannot be greater than 68 characters.

Translation and Validation Tables

- Press Enter to save this information. The Add Validation Table Entries panel (TM04) is displayed. The width of the columns is determined by the values you entered on the previous panel.

TM04 Add Validation Table Entries 1 to 15 of 15

Table ID: XYZCITY Description: Cities with XYZ offices

Entry	Description
Blairsville	Home office
Blue Ridge	South branch
Young Harris	North branch

- In the *Entry* field, enter a validation table entry. In the *Description* field, enter a brief optional description of the entry. Repeat for each entry you want in this table.

Note: Table entries are case sensitive: *abc* is not the same as *ABC*.

When you fill all of the available lines and press Enter, another panel of lines is displayed. If you do not need the extra lines, go to the next step.

- Press Enter to save the table entries. If another panel of lines is displayed, press F3 (Exit).

The Translation and Validation Tables panel (TM01) is redisplayed.

- Press F3 (Exit) to return to the Administrator's Menu (MP01).

Updating a Validation Table

To update a validation table, follow these steps:

- From the Administrator's Menu (MP01), select *Translation and Validation Tables*. The Translation and Validation Tables panel (TM01) is displayed.
- Type **U** in the action column next to the table you want to update, and press Enter.

Add Copy Delete List Print Update View				
TM01		Translation and Validation Tables		1 to 4 of 4
Action	Table ID	Type	Description	
-	ALPHANUM	V	Alphanumeric validation table	
-	CHARSET	V	Character set validation table	
-	TRANS	T	Translation Table	
-	UNITS	T	Convert unit codes to standard	
u	XYZCITY	V	Cities with XYZ offices	

The Update Validation Table Entries panel (TM07) is displayed. You can change the description of the table, the table entries, and the descriptions of the table entries.

Add	Delete	List
TM07	Update Validation Table Entries 1 to 7 of 7	
Table ID: XYZCITY Description Cities with XYZ offices _____		
Action	Entry	Description
-	Blairsville	Home office _____
-	Blue Ridge	South branch _____
-	Young Harris	North branch _____

3. To delete entries, type **D** in the action column next to each entry you want to delete, and press Enter.
4. To display a specific entry at the top of the list, type **L** and the entry on the command line, and press Enter.
5. To add an entry, do the following:
 - a. Type **A** in the action column next to any item, and press Enter.
 - b. A blank line is inserted above that entry. On that line, type the new entry and description.
6. When you are finished updating the table, press F3 (Exit) twice to return to the Administrator's Menu (MP01).

Note: Changes you make to a translation table affect the translation process as soon as the table is saved. This is different than changes to standards, ADFs, and maps, which do not affect translation until the control string is generated again.

Chapter 9. Mapping Your Application Data to an EDI Standard Transaction Set

What is Mapping?	9-1
What is a Trading Partner Transaction?	9-1
Overview of the Mapping Process	9-2
Before You Begin Mapping	9-2
EDI Segments and Data Elements	9-3
Using Accumulators	9-3
Using Literals	9-4
Using Literals for Send Mapping	9-5
Segment Creation for Send Mapping	9-5
Using Literals for Receive Mapping	9-6
Format of Literal Data	9-6
Accumulator Literals	9-6
Conditional Processing Literals	9-7
Literal Keywords	9-8
Named Variables	9-16
Expressions	9-17
Boolean Operators	9-18
Comparison Operators	9-18
Arithmetic Operators	9-18
Unary Operator	9-19
Special Operators	9-20
Date Conversion Special Operators	9-21
Order of Precedence	9-24
DI Variables	9-24
Mapping Techniques for Literal Keywords	9-26
Examples of Using Literal Keywords and Named Variables	9-27
Control Data Literals	9-33
Mapping Specific Service Segment Fields (Receive Only)	9-33
Mapping Generic Service Segment Fields (Receive Only)	9-34
Mapping Service Segment Fields (Send Only)	9-36
Validation During Mapping	9-37
Mapping Guidelines	9-38
Defining a Data Hierarchy for Mapping	9-38
Mapping Nonrepeating Segments	9-39
Mapping Repeating Segments Not in Loops	9-39
Mapping Loops	9-39
Single Occurrences	9-39
Multiple Occurrences	9-40
Repeating Segment Within a Loop	9-40
Mapping Data Elements to and from Structure Fields	9-40
Send Mapping One to Many, Many to One	9-41
Receive Mapping to Multiple Fields	9-41
Mapping Floating Segments	9-41
Processing Bounded Loops	9-42
Preparing the Mapping Worksheet	9-43
Creating a Trading Partner Transaction (Map)	9-45
Defining a Transaction	9-47
Mapping Segments and Data Elements	9-52
Specifying an Application Field Name	9-57

Specifying an Accumulator and Action	9-57
Specifying a Literal	9-58
Special Handling	9-59
Concatenating Data for a Data Element (Send Only)	9-61
Substringing Data for an Application Field (Receive Only)	9-64
Using Named Variables and Special Operators	9-65
Specifying Occurrences	9-67
Qualifying a Segment	9-68
Qualifying a Loop	9-69
Qualifying a Data Element	9-71
Specifying Trading Partners	9-78
Add Trading Partner Usage for Sending	9-78
Trading Partner Usage Overrides for Sending	9-82
Defining Generic Send Usages	9-85
Defining Send Usages	9-85
Determining the Trading Partner	9-86
Selecting the Send Usage	9-86
Standard Envelope Profile Member Selection	9-87
Transaction Store Considerations	9-88
Management Reporting Considerations	9-88
Export/Import Considerations	9-88
Add Trading Partner Usage for Receiving	9-88
Trading Partner Usage Overrides for Receiving	9-94
Defining Generic Receive Usages	9-96
Receive Usage Definition	9-96
Determining the Trading Partner	9-96
Selecting the Receive Usage	9-97
Functional Acknowledgment Processing	9-97
Transaction Store Considerations	9-97
Management Reporting Considerations	9-97
Export/Import Considerations	9-97
Migrating a Transaction Mapping	9-97
Migrating Trading Partner Usages	9-101
Generating Control Strings	9-103
Hierarchical Loops	9-105
The HL Segment	9-106
Preparing Hierarchical Loops	9-107
Literal Keywords for the HL Segment	9-109
Mapping the HL Segment	9-110
Hierarchical Loop Case - Ship Notice (856)	9-116
Hierarchical Loop Case - Ship Notice (856) with a Special Requirement	9-117
DataInterchange Inbound and Outbound Processing Flowcharts	9-118

Chapter 9. Mapping Your Application Data to an EDI Standard Transaction Set

This chapter describes how to map your application data to an EDI standard transaction set. This chapter assumes that you are familiar with your application data layout and have already defined your application data to DataInterchange. It also assumes you are familiar with the EDI standard transaction set you are using.

What is Mapping?

Mapping is the process of telling DataInterchange which fields in your application data format correspond to which data elements in an EDI standard transaction set.

What is a Trading Partner Transaction?

A trading partner transaction consists of a transaction mapping and trading partner usages. The transaction mapping defines how your application data, defined to DataInterchange by your application data format, maps to an EDI standard transaction set. The transaction mapping identifies the transaction set, the data format, the direction (send or receive), and the EDI standard. Because you can use the same transaction mapping for more than one trading partner, each trading partner usage identifies a trading partner that uses the transaction mapping. The trading partner usage can also provide values that are specific to that trading partner, such as:

- Validation and error levels.
- Unique values used during enveloping that override the values in the envelope profile member.
- For receiving transactions, the name of the application file where data should be written, overriding the value in the application data format.

“Specifying Trading Partners” on page 9-78 provides detailed information about all trading partner usage values.

Note: Trading partner transactions (maps) can reside physically in the DataInterchange host database or in the DataInterchange Client database. Mappings which reside in the DataInterchange Client database are identified in the TPT list by special flags. These mappings cannot be updated using the DataInterchange host interface. For more information about these flags, see the Note on 9-48.

See “Using DataInterchange with the DataInterchange Client Graphical User Interface” on page 1-13 for more information about using the DataInterchange Client interface.

Overview of the Mapping Process

Figure 9-1 illustrates the steps in the mapping process.

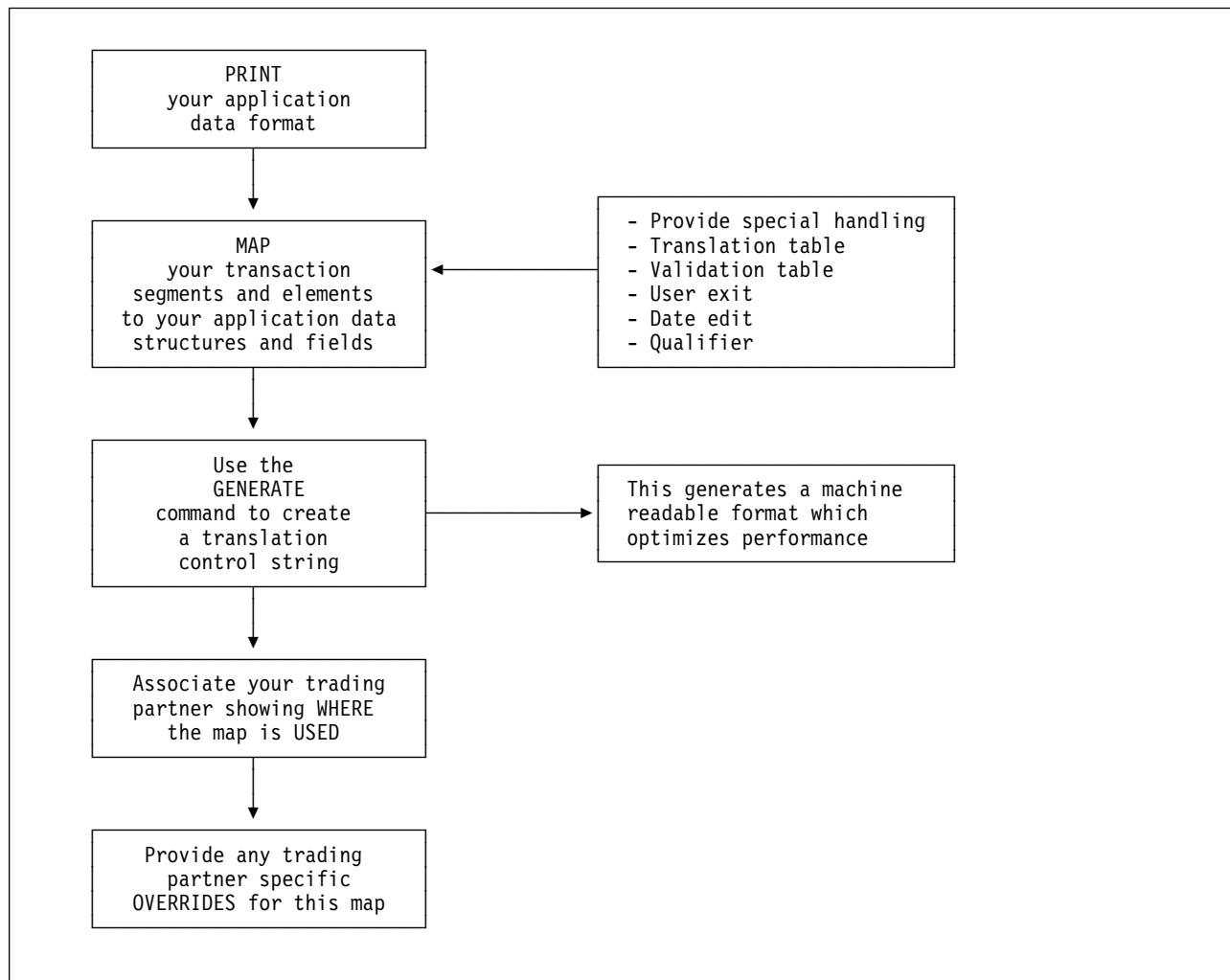


Figure 9-1. Steps in the Mapping Process

Before You Begin Mapping

Before trying to map your application data, prepare a mapping worksheet that shows how your application data format corresponds to the EDI standard transaction set. To do this effectively, you should first understand:

- EDI segments and data elements
- Accumulators and their use
- Literals
- Special handling processing options

EDI Segments and Data Elements

EDI standards specify how many times each segment can be used, and what data elements are associated with each segment.

Transactions are always processed starting with the first data element of the first segment and proceeding to the last segment, as documented in the standard. This is true for both send and receive processing. However, for receive processing, if a segment is not in the input, its associated element instructions are not executed. If a segment is present in the input, all element mappings are executed even if the corresponding element or elements are not present. However, the mapping instructions are executed only if there is a qualifier match in the corresponding input. See “Qualifying a Segment” on page 9-68, “Qualifying a Loop” on page 9-69, and “Qualifying a Data Element” on page 9-71 for information on qualifying segments, loops, and data elements.

When you map your application data fields to the EDI standard data elements, you may have to specify additional information that is not available in your application. For example, an application data field might map to a data element within a segment; however, other data elements within that segment might be required by the standard but are not present in your application data, as shown in the following figure.

ADF Field	Transaction Set Segment and Data Elements

	SEGMENT 1
	DATA ELEMENT 1 (required)
FIELD 1 ----->	DATA ELEMENT 2 (required)
	DATA ELEMENT 3 (required)
	DATA ELEMENT 4 (optional)
	SEGMENT 2
	DATA ELEMENT 1 (optional)
FIELD 2 ----->	DATA ELEMENT 2 (required)
	SEGMENT 3
FIELD 3 ----->	DATA ELEMENT 1 (required)

Figure 9-2. Required Data Elements

Field 1 maps to the second data element of Segment 1, which also requires data elements 1 and 3. Even if your application data does not have corresponding data for these elements, you must somehow specify this data. You do this in several ways:

- Using an accumulator, as described in “Using Accumulators.”
- Using a literal, as described in “Using Literals” on page 9-4.
- Using one of the special handling options, as described in “Special Handling” on page 9-59.

| For more information about EDI standard segments and data elements, see Chapter 6, “Customizing EDI
| and Envelope Standards,” or the information available from your standards organization.

Using Accumulators

You can use accumulators to count occurrences of an event, such as counting the detail line items in a purchase order to provide a hash total. You can also use accumulators to total fields for control purposes, such as totaling the quantity field to cross check the number of items sent or received.

Accumulators can apply to individual transactions or to all transactions in a translation session.

Mapping Your Application Data to an EDI Standard Transaction Set

Valid accumulators are:

Name	Description
T0—T9	Transaction accumulators that apply to one transaction. They are reset at the beginning of each transaction.
G0—G9	Global accumulators that apply to an entire translation session. They are reset at the beginning of each translation session.

Basic accumulator actions are adding, incrementing, zeroing, and mapping. These actions can be combined as follows:

Action	Description
A	For sending, adds to the accumulator the standard data just mapped. For receiving, adds to the accumulator the standard data received. The data must be numeric.
I	Increments the accumulator by 1.
M	Maps the accumulator to the standard data element (send) or the application field (receive).
Z	Zeroes the accumulator.
MI	Maps the accumulator, then increments it.
IM	Increments the accumulator, then maps it.
MZ	Maps the accumulator, then zeroes it.
AM	Adds to the accumulator, then maps it.
MA	Maps the accumulator, then adds to it.

For send transactions, accumulator actions will not be processed unless one of the following occurs:

- Data is generated for the standard field
- The variable is mapped

For receive transactions, accumulator actions will not be processed unless one of the following occurs:

- The data element associated with the accumulator is received
- The accumulator is mapped and at least the segment containing the data element is received.

To map both an accumulator and a received value for the same data element, use the Repeat action to create another mapping occurrence of the data element. Then map one occurrence from the data element to a field and the other occurrence from the accumulator to a field.

Accumulators have the following limitations:

- Each accumulator holds a maximum of 31 digits.
- Each data element mapping can support up to 4 accumulators.
- You can use up to 10 transaction accumulators for a transaction.
- You can use up to 10 global accumulators for an entire translation session.

Using Literals

For send transactions, literals let you supply data that is not in your application data. For receive transactions, literals let you supply data required by your application that is not received with the standard data. DataInterchange offers a variety of options for using literals. This section provides details on each option available, including rules for use, keywords, and syntax.

Transactions are always processed starting with the first data element of the first segment and proceeding to the last data element of the last segment, as documented in the standard. This is true for both send and receive processing.

Using Literals for Send Mapping

When mapping literals for sending data, you can map both an application field and a literal to the same data element. The literal value is used if one of the following conditions occur:

- The structure containing the application data field was provided, but the application data field does not supply a value or contains all blanks.
- The application data field supplies a value that does not match a value in the validation or translation table specified in the mapping.
- The application data field supplies a value, but the conversion from the application data type to the EDI standard data type fails.

DataInterchange attempts a conversion for any application data field defined as a numeric field. The conversion removes leading and trailing blanks, leading zeros before the decimal, trailing zeros after the decimal, and changes the decimal point if the application decimal notation is different from the standard decimal notation. If the data types are numeric, the conversion fails if the data contained anything other than a number or a decimal point. The literal value is used if the conversion fails. See “Validation During Mapping” on page 9-37 for more information.

- If a translation table is specified, the literal value is checked against this table. If no matching entry is found, the translator logs a warning message in the event log, then uses the literal value. For information about the event log, see Chapter 12, “Event Logging.”

Segment Creation for Send Mapping

When an EDI standard segment is mapped with a combination of literal values and application data, the application data determines if the segment is produced. When the values from the application data produce data for the segment, the EDI standard segment is created. When the values in the application data do not produce data for the segment, such as when the structures are not found, the application fields contain all blanks or zeros, or the application field values are not found in the associated validation or translation tables, then the EDI standard segment is not created. An exception is when zeros are passed to a mandatory numeric data element, in which case the segment is created. If this is not desirable, then pass blanks in the application data instead, or use the logic discussed below to suppress the segment.

In some cases it may be desirable to suppress a segment altogether depending on input application data values. It is possible to suppress a segment using **IF** logic. When all contributing data element mappings for a segment contain conditional **IF** statements and all conditions are false, the segment is not produced. A contributing data element mapping is one that may produce output in the corresponding data element; such as specifying an application field name or a literal value, or specifying **USE** on a variable. Noncontributing data element mappings, such as **SAVE** or **SET**, do not have any effect on segment creation. Refer to Table 9-1 on page 9-8 for details on **IF**, **USE**, **SAVE**, and **SET** usage.

Note: If the segment being suppressed begins a loop of segments, the entire loop is suppressed.

Using Literals for Receive Mapping

If a data element mapping provides a literal, the literal value is used if one of the following conditions occur:

- The EDI standard data includes the segment being mapped, but the data element within the segment does not supply a value.
- The EDI standard data element supplies a value that does not match a value in the validation or translation table specified in the mapping.
- The EDI standard data element supplies a value, but the conversion from the standard data type to the application data type fails. DataInterchange attempts a conversion for any EDI standard data field defined as a numeric field. If the data types are numeric and the standard data field contains nonnumeric characters, the conversion would fail and the literal value would be used.
- The EDI standard data element supplies a value, but the &FORCE special literal is used to force the literal into the application field regardless of the standard data element's contents.

Format of Literal Data

When using literals in send or receive mapping:

- For data types BN, Bn, HX, Hn, IT, In, Ln, PD, Pn, ZD, and Zn, the translator converts the literal before placing it in the standard data (send) or the application data (receive). For sending, it converts the literal to character data. For receiving, it converts the literal to the application data type. For example, if the data type is binary, the translator converts the literal to binary, then moves it to the application field.
- Do not type a decimal point when it is implied. For example, if you want to use a default value of 9.99 for a field defined as data type P2 (packed number with two implied decimal positions), enter 999 as the value of the literal.
- Enter literal values for hexadecimal fields as hexadecimal strings. For example, if the application field is defined as a one-byte hexadecimal field and you want to use a default value of X'FF', enter FF as the value of the literal. For receiving, the translator converts each two bytes of literal value to a single byte of application data.
- You can specify a literal value of zero to move a value into the standard field. DataInterchange generally removes leading zeros from an application field so that an application field containing nothing but blanks or zeros will not result in a value for the data element. A value of zero is treated the same as all other literal values when determining if a segment should be created. The &ZEROSIG special literal may also be used to indicate that zeros within the application field are significant.
- In translation and validation tables, enter numeric values left-justified and formatted according to the application data format. For example, if the data format defines a field as R2, enter the value 7 as 7.00 or the value 7.1 as 7.10.

Accumulator Literals

The following literals allow access to the local and global accumulators. You can map an accumulator using the accumulator action, as described in "Using Accumulators" on page 9-3, or you can map it using the literal associated with the accumulator. Using the literal allows a user exit to gain control. An accumulator has a data type of R. The following list describes the literals:

Literal	Description
&Tn	Where <i>n</i> can be 0 through 9, identifies the accumulators T0 through T9.
&Gn	Where <i>n</i> can be 0 through 9, identifies the accumulators G0 through G9.

Conditional Processing Literals

Conditional processing lets you define how you want data processed, based on rules you establish. The following terms will be used in discussing conditional processing:

Term	Description
Named variable	A name used to represent data whose value can be changed while a program is running. You supply the name you want to use.
Expression	A sequence of instructions which can consist of named variables, literals, operators, and constants. When processed, this sequence of instructions provides a single value.
Constant	A value that does not change.
Operator	A symbol that represents an operation to be done. DataInterchange uses arithmetic operators, Boolean operators, comparison operators, unary operators, relational operators, and special operators.
Operation	An action performed on one or more data items such as multiplying, comparing or moving.
Value	A data value you provide. You can use any of the special literals that provide a data value; for example, &DATE, &TIME, &E, &ICN. This includes &T0 through &T9 and &G0 through &G9 to get the values for global and local accumulators.
Default value	A default data value you provide. This value should not be enclosed in quotation (") marks. You can use any of the special literals that provide a data value; for example, &DATE, &TIME, &E, &ICN. This includes &T0 through &T9 and &G0 through &G9 to get the values for global and local accumulators.

Note: When using conditional processing on a data element used to qualify a loop or repeating segment, the first occurrence of the qualifying data element mapping must specify a literal or application data field.

Literal Keywords

Table 9-1 (Page 1 of 9). Literal Keywords

Keyword	S/R	Keyword Description and Syntax
&ACFIELD	S/R	<p>Syntax: &ACFIELD</p> <p>Substitutes the application control value established by the mappings of the AC field from the application data format or the concatenation of the fields specified during creation of the transaction mapping.</p> <p>Note: Literals specified with &LIT and variables specified with &VAR in the mapping are not available in the AC field until the end of the translation process and will not appear in the AC field data that is moved using the &ACFIELD keyword.</p>
&ASSERT n	S/R	<p>Syntax: &ASSERTn(<i>expression</i>)</p> <p>The action associated with this literal is only executed if the <i>expression</i> is false and the assertion level is not greater than n. The &ASSERT keyword can be combined with:</p> <ul style="list-style-type: none"> • A mapping between an application data field and standard data elements • The &SET, &SAVE, and &USE keywords • An error condition; for example, &ASSERTn followed by &ERR. <p>The differences between &ASSERT and &IF are:</p> <ul style="list-style-type: none"> • &ASSERT is a statement about the transaction that is expected to be true. For example, the number of items processed in the transaction should equal the total number of items claimed to be in the transaction (value from CTT segment). If the &ASSERT is not true, special action should take place. An &ASSERT is usually associated with an &ERR condition, but this is not required. <p>With &IF, if the expression is true, special action should take place. An &IF is usually associated with a mapping or named variable, but this is not required.</p> <ul style="list-style-type: none"> • &ASSERT has 10 levels (&ASSERT0 through &ASSERT9). Level 9 assertions are always executed. Assertions at level 0 (&ASSERT0) through 8 are controlled by the assertion level used to start the translation. Thus, it is possible to turn off assertions, but &IF conditions are always checked. <p>When using the API, the assertion level is set in the ASSERTLVL field of the translator control block. When using the DataInterchange Utility, the assertion level is set with the ASSERTLVL keyword on the PERFORM command. See the <i>DataInterchange Programmer's Reference</i> for additional information on the API.</p> <p>See "Expressions" on page 9-17 for more information.</p> <p>For an example of using this keyword, see "Example 8" on page 9-30.</p>
&DATE	S/R	<p>Syntax: &DATE</p> <p>Substitutes the system date. The length of the date field in the standard data (send) or application data (receive) determines whether the date is formatted as <i>yyyymmdd</i> or <i>yymmdd</i>. The date is then edited as requested by the date edit specified in mapping.</p> <p>You can use the &DATE keyword as source data for any of the standard data types.</p> <p>You can also combine the &DATE keyword with the &IFDATA, &IFNODATA, or &FORCE keywords.</p> <p>For an example of using this keyword, see "Specifying a Literal" on page 9-58.</p>

Table 9-1 (Page 2 of 9). Literal Keywords

Keyword	S/R	Keyword Description and Syntax
&DEFERRED	S	<p>Syntax: &DEFERRED &USE <i>variable</i></p> <p>Allows you to signal that at a later time a value will be set using the same name as specified in the &DEFERRED &USE mapping. When the value is set, the data is put into the mapped segment.</p> <p>For example, if you had an HDR segment that contained a total number of items field, you could use &DEFERRED &USE TOTITEMS while mapping the HDR segment field.</p> <p>Note: It is not recommended that the value be set within the same segment mapping. This will produce unexpected results. If the entire segment is mapped based on the use of deferred mapping, the segment could be produced without data.</p> <p>At the end of the mapping, you would then repeat map some field with &SET TOTITEMS X, which will move the literal value 'X' into the HDR segment field. If X is used as a variable, the mapping would be &SET TOTITEMS &E(X), which would move the contents of the variable X into the HDR segment field.</p>
&E	S/R	<p>Syntax: &E(<i>expression</i>)</p> <p>The <i>expression</i> is evaluated and the result used as if it had been entered directly as a literal value. This keyword allows calculations without using a user exit. See “Expressions” on page 9-17 for more information.</p> <p>For examples of using this keyword, see “Example 7” on page 9-30, and the mapping example on page 9-66.</p>
&ERR	S/R	<p>Syntax: &ERR(<i>level,code,facode,text</i>)</p> <p>Allows you to establish your own errors for a transaction. This literal keyword can be used in one of three ways:</p> <ul style="list-style-type: none"> • On the Literal line itself • &IF (<i>expression</i>) • &ASSERT (<i>expression</i>) <p>Note: If you specify a field in the application name field, you cannot use &ERR on this occurrence of the element mapping.</p> <p><i>level</i> is the severity of error where 1=data element, 2=segment, 3=transaction.</p> <p><i>code</i> is the unique error code that should be associated with the error. This value can range from 0 to 999. DataInterchange automatically adds 5000 to separate this value from DataInterchange detected errors.</p> <p><i>facode</i> is the functional acknowledgment error code that should be associated with this error (receive only).</p> <p><i>text</i> is some text that is included in an error message logged by DataInterchange if this error is detected.</p> <p>If an &ERR special literal is executed (normally controlled by either an &IF or an &ASSERTion), then DataInterchange will log message TR0026. Within this message, the <i>text</i> and <i>code</i> will be identified. The <i>level</i> that you assign in the &ERR becomes the extended return code from the translator and thus the JCL condition code in the DataInterchange Utility. If <i>level</i> exceeds the acceptable error level specified in the transaction usage, then the translation will not be successful and the application data will not be returned. The value of <i>code</i> plus 5000 will also be added to the list of errors for the transaction. These are available to the API programs in the ERRCODE field of the translator control block.</p> <p>For an example of using this keyword, see “Example 8” on page 9-30.</p>

Mapping Your Application Data to an EDI Standard Transaction Set

Table 9-1 (Page 3 of 9). Literal Keywords

Keyword	S/R	Keyword Description and Syntax
&FORCE	R	<p>Syntax: &FORCE <i>value</i></p> <p>Forces a literal value into an application field regardless of the standard data element's contents. A literal value specified for receive mapping is normally used only if the standard data element does not contain any data, or if an error occurs while processing the data (see "Using Literals for Receive Mapping" on page 9-6).</p> <p>For the &FORCE keyword to be effective, you must use it in a data element of a segment that is present in the input transaction.</p> <p>For an example of using this keyword, see "Receive Transaction" on page 9-67.</p>
&FORMAT	S/R	<p>Syntax: &FORMAT</p> <p>Substitutes the application data format ID.</p> <p>You can also use the &FORMAT keyword with the &IFDATA, &IFNODATA, or &FORCE keywords.</p>
&IF	S/R	<p>Syntax: &IF(<i>expression</i>)</p> <p>The action associated with this literal is only executed if the <i>expression</i> is true. The &IF literal can be combined with:</p> <ul style="list-style-type: none"> • A mapping between application and standard data fields • A request to SET/SAVE/USE a named variable • An error condition (for example, an &IF followed by &ERR) <p>The differences between &ASSERT and &IF are:</p> <ul style="list-style-type: none"> • &ASSERT is a statement about the transaction that is expected to be true. If the &ASSERT is not true, special action should take place. <p>With &IF, if the expression is true, special action should take place. An &IF is usually associated with a mapping or named variable, but this is not required.</p> <ul style="list-style-type: none"> • &IF conditions are always checked. <p>See "Expressions" on page 9-17 for more information. For an example of using this keyword, see "Example 5a" on page 9-28, "Example 5b" on page 9-29, and "Example 6" on page 9-29.</p>
&IFDATA	S/R	<p>Syntax: &IFDATA <i>value</i></p> <p>For sending, uses a literal value only if an application field contains data. For example, a segment has a pair of data elements for a qualified value and its qualifier. The application data has a corresponding field for the qualified value, but not for the qualifier. You want to supply the qualifier with a literal, but only when the application supplies a qualified value. You can do this using the &IFDATA keyword with the Application field name and Literal fields on the Map Data Element panel (TP10). &IFDATA can be used in combination with &SET or &SAVE keywords.</p> <p>The test performed by the translator to determine whether or not a source field contains data is different depending on the data type of the source field. For numeric data types, zeros are not considered significant, in which case the assumption is made that the field does not contain data. &IFDATA can be used in combination with &ZEROSIG if you want a zero value to be considered significant. &ZEROSIG must precede the save keyword.</p> <p>&IFDATA is allowed on receive but only when used in combination with the &SET or &SAVE keywords.</p> <p>For an example of using this keyword, see "Example 2" on page 9-27.</p>

Table 9-1 (Page 4 of 9). Literal Keywords

Keyword	S/R	Keyword Description and Syntax
&IFNODATA	S/R	<p>Syntax: &IFNODATA <i>value</i></p> <p>Uses a literal value only if an application field does not contain data. If you do not use a keyword before the literal, &IFNODATA is used by default.</p> <p>&IFNODATA can be used in receive transactions, but only when used in combination with the &SET keyword.</p> <p>For an example of using this keyword, see “Specifying a Literal” on page 9-58.</p>
&IFNOVAR	S/R	<p>Syntax: &IFNOVAR</p> <p>This keyword is only valid when used with &SAVE, &LSAVE, &SET, or &LSET:</p> <p style="padding-left: 40px;">&IFNOVAR &SAVE <i>variable</i> &IFNOVAR &LSAVE <i>variable</i> &IFNOVAR &SET <i>variable</i> &IFNOVAR &LSET <i>variable</i></p> <p>The named variable will only be created if it does not already exist. If the named variable already exists, the data in it is not overlaid.</p> <p>For an example of using this keyword, see “Example 3” on page 9-27.</p>
&LOOPBREAK	S	<p>Syntax: &LOOPBREAK</p> <p>Use when an outer and an inner loop are qualified on the same structure, and when you want the inner loop to be generated more than once. By putting a &LOOPBREAK in the inner loop, then the inner loop will be repeated until the &LOOPBREAK condition is met. The &LOOPBREAK condition is established using &IF; for example,</p> <p style="padding-left: 40px;">&IF((A < B) AND (X > Y)) &LOOPBREAK</p> <p>Note: You cannot enter an Application field name on the same mapping occurrence that uses &LOOPBREAK.</p>
&LOOPCHECK	S	<p>Syntax: &LOOPCHECK</p> <p>This keyword is very similar to &LOOPBREAK, but &LOOPCHECK does all the conditional processing automatically, based on the application field name used in the mapping that specifies &LOOPCHECK. DataInterchange will save this field value the first time the loop is created, and will continue to create inner loops until a record with a different non-blank field value is found.</p> <p>Note: You cannot enter an Application field name on the same mapping occurrence that uses &LOOPCHECK.</p>

Mapping Your Application Data to an EDI Standard Transaction Set

Table 9-1 (Page 5 of 9). Literal Keywords

Keyword	S/R	Keyword Description and Syntax
&LSAVE	S/R	<p>Syntax: &LSAVE <i>variable</i><,<<i>position</i>><,<<i>length</i>> <<i>default value</i>></p> <p>Saves the value from the current data element in a named variable, but only for the duration of the loop instance. A value saved in an outer loop is available within associated inner loops. A repeating segment is considered to be a loop; therefore, values saved in one instance of a repeating segment are not available in subsequent segment iterations.</p> <p>The value to be saved is normalized prior to being stored in the variable. The normalization that takes place depends on the data type of the source field being saved. For character data types A, AC, AN, CH, or ID, any trailing blanks are removed from the source value. If the value in the field (defined with a character data type) contains all numbers, the data will be treated as a numeric data type. Leading zeros will be removed from numeric data types. For numeric data types, the source value has all leading and trailing blanks removed, it is converted to a REAL value, and all leading zeros before the decimal and trailing zeros after the decimal are removed. If a source field is numeric and contains a zero value, the variable will contain null. If a value of zero needs to be saved, &ZEROSIG must be used in combination with the save. &ZEROSIG must precede the &LSAVE keyword.</p> <p><i>position</i> is optional, but if provided, it indicates the position within the current variable where the information should be saved. You can use an asterisk (*) to save at the end of the variable; for example, &LSAVE name,*. If <i>position</i> is not specified, the data replaces the current value of the variable.</p> <p><i>length</i> is optional, but if provided, it indicates the length of data that should be saved. You can use an asterisk to save the total length of the data being supplied; for example, &LSAVE name,*,*.</p> <p>For example, if you want data to be saved starting in position 4 and the data you want saved is 5 characters long, the position and length would be stated as 4,5. Stating both the position and length, you can combine more than one data element into a single named variable. Special processing may be needed when combining data elements with different data types into a single named variable. See “Example 9” on page 9-31 and “Example 10” on page 9-32.</p> <p>The <i>default value</i> is also optional, and if provided, the value is saved in the <i>variable</i> when the current data element contains no value.</p> <p>If the named variable does not already exist, it is created. If it already exists, the data in the existing variable is overlaid with the new data. If the data element is empty and no default value is supplied, the named variable is created but it contains no data.</p> <p>A variable established with &LSAVE does not affect the value of a variable with the same name at any other looping level. If the variable was created outside the loop with LSAVE or SAVE, and the variable is created again inside the loop, the USE for the variable inside the loop will use the value which was saved inside the loop. If the USE for the variable is outside the loop, then the value which was saved outside the loop will be used.</p> <p>For an example of using this keyword, see “Example 3” on page 9-27.</p> <p>When zeros are passed in a DT ADF field and mapped to a DT element, they are considered significant data, and are populated to the element. In other words, DataInterchange treats DT data as character versus numeric.</p>

Table 9-1 (Page 6 of 9). Literal Keywords

Keyword	S/R	Keyword Description and Syntax
&LSET	S/R	<p>Syntax: &LSET <i>variable</i><,<i>position</i>><,<i>length</i>> <i>value</i> The named variable is created or overlaid with the stated value, but only for the duration of the loop instance. A value set in an outer loop is available within associated inner loops. A repeating segment is considered to be a loop; therefore, values set in one instance of a repeating segment are not available in subsequent segment iterations.</p> <p><i>position</i> is optional, but if provided, it indicates the position within the current variable where <i>value</i> should be set. You can use an asterisk to set <i>value</i> at the end of the variable, for example, &LSET <i>name</i>,* <i>abcd</i>. If <i>position</i> is not specified, <i>value</i> replaces the current value of the variable.</p> <p><i>length</i> is optional, but if provided, it indicates the length of <i>value</i> that should be set. You can use an asterisk to set the total length of <i>value</i>, for example, &LSET <i>name</i>,*<i>abcd</i>. Special processing may be needed when combining data elements with different data types into a single named variable. See “Example 9” on page 9-31 and “Example 10” on page 9-32.</p> <p>A variable established with &LSET does not affect the value of a variable with the same name at any other looping level. If the variable was created outside the loop with LSET or SET and the variable is created again inside the loop, the USE for the variable inside the loop will use the value which was set inside the loop. If the USE for the variable is outside the loop, then the value which was set outside the loop will be used.</p> <p>When zeros are passed in a DT ADF field and mapped to a DT element, they are considered significant data, and are populated to the element. In other words, DataInterchange treats DT data as character versus numeric.</p>
&LSID	R	<p>Syntax: &LSID <i>value</i></p> <p>Identifies the instance of the LS loop, where <i>value</i> is equal to the LS01 value in the standard data. This special literal is required only if the translator has no other way to determine which LS loop is provided. See “Processing Bounded Loops” on page 9-42 for more information.</p>
&SAMEAS	S/R	<p>Syntax: &SAMEAS <i>segno</i></p> <p>Indicates when a mapping for a current data element should be exactly the same as the data element identified by <segno>.</p> <p>For example, if the mapping for element 2 of the POC segment should be exactly the same as the mapping for element 1 of the POC segment, then specify the special literal value of '&SAMEAS 1' when mapping element 2. This results in the mapping for element 2 being the same as element 1, which is useful when qualifying (Q/S) data elements on a receive mapping.</p>

Mapping Your Application Data to an EDI Standard Transaction Set

Table 9-1 (Page 7 of 9). Literal Keywords

Keyword	S/R	Keyword Description and Syntax
&SAVE	S/R	<p>Syntax: &SAVE <i>variable</i><,<i>position</i>><,<i>length</i>> <<i>default value</i>></p> <p>Saves the value from the current data element (inbound) or application field (outbound) in a named variable. The value to be saved is normalized prior to being stored in the variable. The normalization that takes place depends on the data type of the source field being saved. For character data types A, AC, AN, CH, or ID, any trailing blanks are removed from the source value. For numeric data types, the source value has all leading and trailing blanks removed, it is converted to a REAL value, and all leading zeros before the decimal and trailing zeros after the decimal are removed. If a source field is numeric and contains a zero value, the variable will contain null. If a value of zero needs to be saved, &ZEROSIG must be used in combination with the save. &ZEROSIG must precede the &SAVE keyword.</p> <p><i>position</i> is optional, but if provided, it indicates the position within the current variable where the information should be saved. You can use an asterisk to save at the end of the variable; for example, &SAVE name,*. If <i>position</i> is not specified, the data replaces the current value of the variable.</p> <p><i>length</i> is optional, but if provided, it indicates the length of data that should be saved. You can use an asterisk to save the total length of the data being supplied, for example, &SAVE name,*,*.</p> <p>For example, if you want data to be saved starting in position 4 and the data you want saved is 5 characters long, the position and length would be stated as 4,5. Stating both the position and length, you can combine more than one data element into a single named variable. Special processing may be needed when combining data elements with different data types into a single named variable. See “Example 9” on page 9-31 and “Example 10” on page 9-32.</p> <p>The <i>default value</i> is also optional, and if provided, the value is saved in the <i>variable</i>.</p> <p>If the named variable does not already exist, it is created. If it already exists, the data in the existing variable is overlaid with the new data. If the data element is empty and no default value is supplied, the named variable is created.</p> <p>For examples of using this keyword, see “Example 1” on page 9-27, and the examples starting on page 9-65.</p> <p>When zeros are passed in a DT ADF field and mapped to a DT element, they are considered significant data, and are populated to the element. In other words, DataInterchange treats DT data as character versus numeric.</p>
&SET	S/R	<p>Syntax: &SET <i>variable</i><,<i>position</i>><,<i>length</i>> <i>value</i></p> <p>The named variable is created or overlaid with the stated value. If no default value is specified, &SET clears the variable.</p> <p><i>position</i> is optional, but if provided, it indicates the position within the current variable where <i>value</i> should be set. You can use an asterisk to set <i>value</i> at the end of the variable, for example, &SET name,* abcd. If no default value is specified, &SET clears the variable.</p> <p><i>length</i> is optional, but if provided, it indicates the length of <i>value</i> that should be set. You can use an asterisk to set the total length of <i>value</i>; for example, &SET name,*,* abcd. Special processing may be needed when combining data elements with different data types into a single named variable. See “Example 9” on page 9-31 and “Example 10” on page 9-32.</p> <p>When zeros are passed in a DT ADF field and mapped to a DT element, they are considered significant data, and are populated to the element. In other words, DataInterchange treats DT data as character versus numeric.</p>

Table 9-1 (Page 8 of 9). Literal Keywords

Keyword	S/R	Keyword Description and Syntax
&THANDLE	R	<p>Syntax: &THANDLE</p> <p>Substitutes the DataInterchange archive key. Can be used to assist in mapping the SAP IDOC. It enables mapping of the DataInterchange archive key to the SAP IDOC for inbound processing. The length of the THANDLE field is 20 characters and is formatted as YYYYMMDDHHMMSSnnnnnn. It is the concatenation of the date, time, and a sequence number to ensure uniqueness.</p>
&TIME	S/R	<p>Syntax: &TIME</p> <p>Substitutes the system time. The length of the time field in the standard data (send) or application data (receive) determines whether the time is formatted as <i>hhmm</i> or <i>hhmmss</i>.</p> <p>You can use the &TIME keyword as source data for any of the standard data types.</p> <p>You can also combine the &TIME keyword with the &IFDATA, &IFNODATA, or &FORCE keywords.</p>
&TPID	S/R	<p>Syntax: &TPID</p> <p>Substitutes the value of the internal trading partner ID.</p> <p>You can also use &TPID keyword with the &IFDATA, &IFNODATA, or &FORCE keywords.</p>
&TPNICKN	S/R	<p>Syntax: &TPNICKN</p> <p>Substitutes the value of the trading partner nickname.</p> <p>You can also use &TPID keyword with the &IFDATA, &IFNODATA, or &FORCE keywords.</p>
&USE	S/R	<p>Syntax: &USE <i>variable</i><,<position><,length> <default value></p> <p>For inbound transactions, the value of the named variable is the source of data for the application field. The value of the standard data element being mapped is ignored.</p> <p>For outbound transactions, the value of the named variable is used to provide data for the standard data element being mapped. An application field cannot be specified.</p> <p>The named variable must be saved or set using the appropriate keyword before you can use it with this keyword.</p> <p><i>position</i> is optional, but if provided, it indicates the position within the variable from which the data should be retrieved.</p> <p><i>length</i> is optional, but if provided, it indicates the length of data that should be retrieved. You can use an asterisk to move all data beginning at the location specified by the position parameter through the end of the variable, for example, &USE var 3,*.</p> <p><i>default value</i> is optional, and if provided, the default value is used when the variable contains no value.</p> <p>Note: For numeric elements, a variable value of zero causes DataInterchange to use the default value. In the case where no default is specified, as in literal = &USE X, there will be no output if variable X contains zero. At times, however, zero needs to be considered significant. In these cases, the user should specify the default, as in literal = &USE X 0.</p> <p>For an example of using this keyword, see “Example 1” on page 9-27.</p>

Mapping Your Application Data to an EDI Standard Transaction Set

Table 9-1 (Page 9 of 9). Literal Keywords

Keyword	S/R	Keyword Description and Syntax
&ZEROSIG	S	Syntax: &ZEROSIG <default value> Use the keyword &ZEROSIG to indicate that a zero in an application field is significant when mapping to optional or conditional data elements. Without use of this special literal, DataInterchange considers a zero value being mapped to an optional or conditional data element as insignificant and produces no output during translation. You can combine the use of &ZEROSIG with &IFDATA, &IFNODATA, &SAVE, and &LSAVE keywords. &ZEROSIG must precede the other keyword. &ZEROSIG can also be used with a default literal to indicate that the application field is used if it contains a value, including zero, but the default literal is used if the field is blank. Binary data types preclude this because blanks represent real values.

Named Variables

For receiving transactions, one way to use data in one data element for multiple application fields is to map the data element to each application field, using the repeat mapping capability. The disadvantage is that the application structures are always created, whether or not they are needed. Sometimes, you do not want the structure to be created unless some other data within the transaction is present. Named variables let you save the value of a data element until the time when the application structure should be created. Then when you are mapping the data element that creates the structure, you can use the repeat mapping capability to map the value in the named variable.

Named variables are also critical to the use of expressions and conditional processing. See “Expressions” on page 9-17 and “Conditional Processing Literals” on page 9-7 for additional information. Data values from an application field for outbound processing or standard data elements for inbound processing are not directly available for use in an expression. The values must first be saved to a named variable using for example, the &SAVE literal, then the named variable can be used within the expression.

A variable name can be the same as your application field name, up to 16 characters, but it cannot start with any of the following:

- A numeric digit (0 through 9).
- The letter P. These variables are reserved for future use.
- The letters DI. These variables are reserved for DataInterchange.
- An ampersand (&), so they do not get confused with special literals.
- A left parenthesis, so they do not get confused with the start of an expression.

A variable name cannot contain any of the special characters designated as Arithmetic Operators or Alternate Comparison Operators. Use of these special characters within a variable name will produce unpredictable results.

The first character of a variable name determines the life span or scope of the variable:

- If the first letter is anything other than G, the variable has transaction scope. The variable is deleted after the transaction is translated. This is the same scope as local accumulators (T0 through T9). For more information, see “Using Accumulators” on page 9-3.
- If the first letter is G, the variable has translator session scope. The variable is not deleted until the session with the translator is terminated. This is the same scope as global accumulators (G0 through G9). For more information on accumulators, see “Using Accumulators” on page 9-3.
- To create a variable that will exist only for the duration of the loop in which it was created, use the &LSAVE or &LSET literal keywords. When the loop repeats or terminates, any variable created using

these keywords is deleted. A variable established with &LSAVE or &LSET does not disturb the value of a variable with the same name at any other looping level.

Variable names are not case sensitive. *TOTALITEMS* and *totalitems* are the same variable. Special processing may be needed when combining data elements with different data types into a single named variable. See “Example 9” on page 9-31 and “Example 10” on page 9-32.

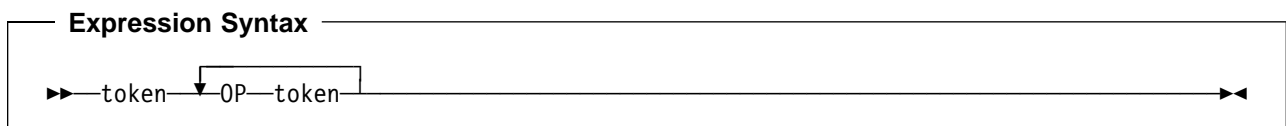
You can combine the substring capability in mapping with the substring capability of named variables. For example, you can use sub strings to get the first 4 bytes of a standard data element, then use the *position* or *length* options to put the data in a specific place in the named variable.

Expressions

Special literals that use expressions are:

- &IF
- &ASSERT
- &E

The syntax for an expression is:



Where *token* can be either:

- A named variable, such as TOTALS
- A numeric constant, such as 1024
- A text constant, such as 'ABCD'

Note: You must use quotation marks around text constants to distinguish them from named variables. You can use single or double quotes (' or "), but whichever is used to start the text constant must also be used to end it.

and where *OP* is one of the following types of operators:

- Boolean
- Comparison
- Arithmetic
- Unary
- Special

By default, the data type of a variable is implicitly assigned. If the contents are all numeric digits, the data type is assumed to be numeric and is treated as such in subsequent comparisons. This assignment does not take into consideration the data type of the source data field that originally supplied the value for the variable. It is based simply on the contents of the variable. If a variable contains any nonnumeric characters, the variable data type is assumed to be character. You can override this implicit data type assignment with the use of the CHAR or NUMBER operator. For more information, see the operators CHAR on page 9-19 and NUMBER on page 9-19.

Boolean Operators

DataInterchange has two Boolean operators: AND and OR. AND returns a value of 1 if both conditions are true, or a value of 0 if either condition is not true. OR returns a value of 1 if either condition is true, or a value of 0 if both conditions are not true. AND and OR must be entered in uppercase because they are case sensitive.

Comparison Operators

Comparison operators tell DataInterchange to compare two objects.

DataInterchange processes comparison operators the same way it does Boolean operators. If the comparison is true, it returns a value of 1. If it is not true, it returns a value of 0.

DataInterchange has the following comparison operators:

Operator	Alternate	Description
EQ	=	The first value is the same as the second value.
GE	>=	The first value is greater than or equal to that of the second value.
GT	>	The first value is greater than the second value.
LE	<=	The first value is less than or equal to that of the second.
LT	<	The first value is less than the second value.
NE	!=	The values are not equal, because they are either different data types or different values.

Notes:

1. If you use the alphabetic operators, such as EQ, they must be uppercase.
2. Comparing a numeric constant or variable to a character constant or variable results in an invalid comparison. This produces a false result for all comparisons with the exception of Not Equal (NE or !=), in which case a true result is returned.
3. If you enclose a numeric constant in quotation marks ("), a numeric data type is still assumed for the data. Therefore, comparison to a variable that contains numeric data will not fail.

Arithmetic Operators

In the following descriptions, single quote marks (') surround absolute values. DataInterchange has the following arithmetic operators:

Operator	Description
+	$a + b$ has a value equal to the sum of the two. For numeric values, the sum is numeric. For example, $2 + 2 = 4$. For nonnumeric values, the sum is a concatenation of the values. For example, 'AB' + 'CD' equals 'ABCD'. The use of quotation marks around a numeric constant has special meaning when used during addition. The enclosed value will be treated as character data and concatenated to the second value. For example, '19' + 940323 = 19940323.
-	$a - b$ has a value equal to the difference between the two. For numeric values, the result is numeric. For example $4 - 2 = 2$. For nonnumeric values, the result is a deconcatenation of the two values. For example, 'ABCD' - 'CD' equals 'AB'. The use of quotation marks around a numeric constant has special meaning when used

during subtraction. The enclosed value will be treated as character data and deconcatenated from the second value. For example, $19940323 - '23' = 199403$.

*	$a * b$ has a value equal to the multiplication of a and b , if both a and b are numeric. For example, $4 * 4 = 16$. If either variable is nonnumeric, the result is created by concatenating the value of the b after each character in a . For example, 'ABCD'*'Z' equals 'AZBZCZDZ'.
/	a / b has a value equal to the division of a by b , if both a and b are numeric. For example, $16 / 4 = 4$. Division by 0 yields a 0 value. If either variable is nonnumeric, the result is created by removing each occurrence of b from a . For example, 'ACDBCD'/'CD' equals 'AB'.
CHAR	The CHARacter operator forces DataInterchange to treat a value as a character value rather than a numeric value.
NOT	An operator to reverse the Boolean value of an expression. The exclamation point (!) is a short hand for the NOT operator. NOT(value) has the following meanings: <ol style="list-style-type: none"> 1. If value is numeric, then NOT(value) is 1 if the value is 0 and 1 otherwise. Thus, NOT(0) yields 1 and NOT(1) yields 0. 2. If value is a string value, then NOT(value) is 1 if the string has no length and zero otherwise. Thus, NOT('abc') yields 0 and NOT('') yields 1.
NUMBER	The NUMBER operator forces DataInterchange to treat a value as a numeric value rather than a character value. DataInterchange will normally treat a value in quotes as a character value and, if both operands look like character values then what might be thought of as a numeric operator will be treated as a string operator. Thus, the expression &E('12' + '34') will yield '1234' because both operators are flagged as character data. If what you really wanted was an arithmetic addition rather than a string concatenation then you could use &E(NUMBER('12') + NUMBER('34')) which will yield 46 (as would &E(12 + 34)).
RD or :	$a \text{ RD } n$ has a value equal to a rounded to n decimal places if a is numeric. For example, $4321.556 \text{ RD } 2$ equals 4321.56. If a is nonnumeric, the result is created by taking the first n characters from a . For example, 'ACDBCD' RD 3 equals 'ACD'. If n is less than or equal to zero it is interpreted to be a request to remove leading blanks. Thus, 'ABC' RD 0, yields 'ABC'. If using the character string RD for rounding instead of the special character :, it must be entered using uppercase.
TU or ;	$a \text{ TU } n$ has a value equal to a truncated to n decimal places if a is numeric. For example, $4321.556 \text{ TU } 2$ is 4321.55. Notice that the number 6 is dropped from the result and not rounded. If a is nonnumeric, the result is created by taking the last n characters from a . For example, 'ACDBCD' TU 3 yields 'BCD'. If n is less than or equal to zero it is interpreted to be a request to remove trailing blanks. Thus, 'ABC ' TU 0 yields 'ABC'. If using the character string TU instead of the special character ;, it must be entered using uppercase.

Unary Operator

The following describes the unary (single component) operator:

Operator	Description
-	&E(- a) changes the sign of a . If a does not exist, the value is 0.

Special Operators

The following list describes the special operators:

Operator	Description
UE	<p>&E(<i>a</i> UE 'MYPROG') returns a value from the user-written program MYPROG.</p> <p>Note: As mentioned under the descriptions of &LSAVE on page 9-12 and &LSET on page 9-13, variables are normalized to REAL value equivalents. When using this special operator, the variable value that is passed is converted back to its <i>Nn</i> numeric format.</p>
TS	<p>&E(<i>a</i> TS 'MYTABL') translates the local value <i>a</i> to the standard value using the MYTABL translation table.</p> <p>Notes:</p> <ol style="list-style-type: none">1. As discussed under the descriptions of &LSAVE on page 9-12 and &LSET on page 9-13, variables are normalized to REAL value equivalents. When using this special operator, the variable value that is passed is converted back to its <i>Nn</i> numeric format.2. If the translation table specified does not exist, or the value passed does not match an entry in the table, then no data is produced from this instruction and no errors or exceptions are issued.
TL	<p>&E(<i>a</i> TL 'MYTABL') translates the standard value <i>a</i> to the local value using the MYTABL translation table.</p> <p>Notes:</p> <ol style="list-style-type: none">1. As discussed under the descriptions of &LSAVE on page 9-12 and &LSET on page 9-13, variables are normalized to REAL value equivalents. When using this special operator, the variable value that is passed is converted back to its <i>Nn</i> numeric format.2. If the translation table specified does not exist, or the value passed does not match an entry in the table, then no data is produced from this instruction and no errors or exceptions are issued.
IN	<p>&E(<i>a</i> IN 'MYTABL') returns a value equal to 1 if <i>a</i> exists in the MYTABL validation table.</p> <p>Notes:</p> <ol style="list-style-type: none">1. As discussed under the descriptions of &LSAVE on page 9-12 and &LSET on page 9-13, variables are normalized to REAL value equivalents. When using this special operator, the variable value that is passed is converted back to its <i>Nn</i> numeric format.2. If the validation table specified does not exist, or the value passed does not match an entry in the table, then no data is produced from this instruction and no errors or exceptions are issued.
SC	<p>&E(<i>a</i> SC <i>max.dec</i>) scales a real number to a maximum of <i>max</i> digits with a maximum of <i>dec</i> decimal places, truncating unused digits. For example:</p> <p>&E(1234.56 SC 4.2) yields 1234 &E(1234.56 SC 6.2) yields 1234.56 &E(1234.56 SC 6.1) yields 1234.5 &E(1234.56 SC 8.1) yields 1234.5 &E(1234.56 SC 8.8) yields 1234.56 &E(1234.56 SC 3.3) yields 1234.56 &E(1234.56 SC 4.5) yields 1234</p>

- If *max* is less than the number of significant digits to the left of the decimal point, the SC is ignored.
- If *dec* is greater than *max*, *dec* is set equal to *max*.

SC applied to strings provide a substring capability. When applied to strings, the format is &E('STRING' SC *pos.len*). For example:

```
&E('ABCDEF' SC 4.2) yields 'DE'
&E('ABCDEF' SC 1.5) yields 'ABCDE'
&E('ABCDEF' SC 9.1) yields ''
&E('ABCDEF' SC 1.9) yields 'ABCDEF'
&E('ABCDEF' SC .1) yields 'A'
&E('ABCDEF' SC 5) yields 'E'
```

If you do not provide *pos* or *len*, DataInterchange uses 1 for that value.

SR &E(*a* SR *max.dec*) scales a real number to a maximum of *max* digits with a maximum of *dec* decimal places, rounding the value. For example:

```
&E(1234.56 SR 4.2) yields 1235
&E(1234.56 SR 6.2) yields 1234.56
&E(1234.56 SR 6.1) yields 1234.6
&E(1234.56 SR 8.1) yields 1234.6
&E(1234.54 SR 8.1) yields 1234.5
```

IS &E(*a* IS *pattern*) has a value equal to *a* but establishes a *pattern* for the data within *a*. This *pattern* only has meaning when using the TO operator (next). All variables have a default pattern of "ABCDEFGHJKLMNOPQRSTUVWXYZ".

TO &E(*a* TO *pattern*) has a value that is created by matching the pattern associated with *a* with the *pattern* in this expression. A character from TO *pattern*, if located in the IS *pattern* and if found the corresponding character from *a*, is moved to the result field. If a match cannot be found, the character from the TO *pattern* is moved to the result field. For example:

- To reverse a string: &E('PLEH' IS 'ABCD' TO 'DCBA') yields 'HELP'.
- To insert delimiters: &E('HHMM' IS 'ABCD' TO 'AB:CD') yields 'HH:MM'.
- To remove delimiters: &E('HH:MM' IS 'ABCDE' TO 'ABDE') yields 'HHMM'.

The last three examples all show the use of both the IS and TO operators for clarity. The IS operator is not really necessary because all variables automatically have the default pattern described in the IS operator. Therefore:

- &E('PLEH' TO 'DCBA') yields 'HELP'
- &E('THISAMEG ' TO 'ABCDICDIEIFGDDEHG') yields 'THIS IS A MESSAGE'.

Date Conversion Special Operators

DataInterchange allows for any-to-any date conversions. The format of the any-to-any date conversion operator is:

```
&E(variable FD mask TD mask)
```

where:

variable

The value to be converted.

Mapping Your Application Data to an EDI Standard Transaction Set

FD

From Date operator which signals that the following variable establishes the mask that describes the date within *variable*.

TD

To Date operator which signals that the following variable establishes the mask that describes the date that is wanted.

mask

The mask that describes the FROM or TO date. A mask consists of the following symbols which identify the date provided or date wanted. Symbols may be in upper or lower case. Any value in the mask that is not one of the symbols below is expected to be physically part of the source data (FD) or will become physically part of the result data (TD).

CC Century

YY Year

MM Month of year

DD Day of month

D Day of month as a single character, if possible

HH Hour of day

MM Minute of hour

II Minute of hour

MM can be used when it immediately follows HH as in HHMM; however, if you want minute followed by hour, you must use IIHH, because MMHH would be interpreted as month of year and Hour of day.

SS Second of minute

WW Week of Year (1 through 52)

K Day of Week (Monday=1, Tuesday=2, etc.)

D can be used if it immediately follows WW as in WWD; however, if you want day of week followed by week, you must use KWW, because DWW would be interpreted to be day of month and Week of year.

JJJ Julian day of year

Q Quarter (1,2,3,4)

E Semester (1,2)

ZZZ Time zone

TM Textual month (i.e., January, February, etc.)

TM may be followed by the name of a translate table to convert a textual month to a numeric month (FD), or from a numeric month to a textual month (FD). If a table name is not provided, the default table names are DIMONTXT to translate from text to numeric and DIMONNUM to translate from numeric to text. A table name is indicated using parenthesis, for example; TM(*tablename*), where *tablename* must be a constant.

After processing the From Date and before creating the To Date, the following processing will be done.

1. SS (seconds) will be defaulted to 0.
2. CC (century) will be defaulted to 19 when the YY (year) is greater than 10 and to 20 otherwise.

3. JJJ (julian day) will be created based on WW (week of year) and K (day of week) if not otherwise provided and WW and K were provided.
4. JJJ (julian day) will be created based on MM (month of year) and DD (day of month) if not otherwise provided and MM and DD were provided.
5. JJJ (julian day) will be used to determine WW (week of year), K (day of week), if either WW or K was not provided.
6. JJJ (julian day) will be used to determine MM (month of year), DD (day of month), if either MM or DD was not provided.
7. Q (Quarter) will be determined based on MM (month of year) if not otherwise provided.
8. E (Semester) will be determined based on MM (month of year) if not otherwise provided.

The following are some examples using CONSTANTS for all values.

- Simple example to remove delimiters.

```
&E('96/06/07' FD 'YY/MM/DD' TD 'YYMMDD') yields '960607'
```

- Simple example to remove delimiters and rearrange

```
&E('96/06/07' FD 'YY/MM/DD' TD 'MMDDYY') yields '060796'
```

```
&E('96/06/07 EDT' FD 'YY/MM/DD ZZZ' TD 'ZZZ MMDDYY') yields 'EDT 060796'
```

- Change delimiters and convert from one form to another

```
&E('96/06/07' FD 'YY/MM/DD' TD 'YY:JJJ') yields '96:157'
```

- Textual month to Numeric month

```
&E('June 7, 1996' FD 'TM D, CCYY' TD 'YYMMDD') yields '960607'
```

- Numeric month to Textual month

```
&E('060796' FD 'MMDDYY' TD 'TM D, CCYY') yields 'June 7, 1996'
```

- Numeric month to Textual month with a special translate table that uses abbreviations for the months

```
0 FD 'MMDDYY' TD 'DDTM(ABBREV)CCYY') yields '07JUN1996'
```

Newer releases of X12 and EDIFACT standards contain segments with variable date/time formats. The format is determined by a qualifier value in the segment. DataInterchange provides two tables for dynamically translating the qualifier into a mask. Table EDIXDTMSK is used for X12 and table EDIEDTMSK is used for EDIFACT.

Assume the following X12 data is received in the DTP segment (Date/Time period):

- DTP**D8*19940927!

Where D8 is the date/time period format qualifier (CCYYMMDD) and 19940927 is the date/time period.

Assume your application requires the date in YYMMDD format. You map the DTP segment as follows:

- DTP03 - &SAVE Qual

Results in Qual is D8

- DTP03 - &SAVE Date

Results in Date is 19940927

- DTP03 - &FORCE &E(Date FD (QUAL TS 'DIXDTMSK') TD 'YYMMDD')

Results of (Qual TS 'DIXDTMSK') is CCYYMMDD

Results of &FORCE is 940927

Order of Precedence

During processing, all expressions are evaluated from left to right. The order of precedence is:

1. Unary minus (–)
2. Rounding (RD) and truncating (TU)
3. Special operators (UE, TS, TL, IN, IS, TO, SC, SR, FD, TD)
4. Multiply (*), Divide (/)
5. Addition (+), Subtraction (-)
6. Relational Operators (GT, GE, LT, LE, EQ, NE)
7. Boolean (AND)
8. Boolean (OR)

Precedence can be overridden with parentheses embedded within an expression. For example, $\&E(2+3*5)$ equals 17 because the multiplication is done first, then the addition. $\&E((2+3)*5)$ equals 25 because the parentheses indicate that the addition is done first, then the multiplication.

DI Variables

DataInterchange has reserved the prefix DI for variables that will be reserved for use by DataInterchange to accomplish special functions. The following DI variables are currently available and are used with the **&SET** keyword:

DIAPPPFILE	This variable may be used to change the name of the file to which the translation application data will be written during a Receive Translate. It will override any value that was used in the receive usage or in the application data format definition. It provides the capability for data that is being received to influence the final destination for the data. For example, the statement <code>&SET DIAPPPFILE SPECIAL</code> would force the current transaction to be written to the application file identified by the ddname SPECIAL.
DIAPPTYPE	This variable sets the application file type that corresponds with the file name provided by DIAPPPFILE.
DIAUTOCC	Allows automatic century manipulation for both inbound and outbound translation. Century will be automatically added or removed from the date using the length of the standard data element or application field. For example, the following statement uses a value of 1: <code>&SET DIAUTOCC 1</code>
DICCTRL	Use to remove the century control year from the translator and allow selection of the century control year. If year is greater than 10, century is 19; otherwise century is 20. The century control year is 10. For example, if year is less than 95, century is 20, control year is 95: <code>&SET DICCTRL 95</code>
DICUSERDATA	This variable is used to set the data value that will be inserted in the TRCB field <i>cuserdata</i> . This field is copied to any output 'C' record before the 'C' record is written. Received data can be placed into a named variable in any combination up to 256 bytes. Then use the reserved variable DICUSERDATA anytime the value of the named variable needs to be placed into the TRCB. For example, suppose a named variable <i>tvar</i> has been created and filled with the data from a previously mapped data element.

That data can be placed in the TRCB *cuserdata* field by including the following in a map:

```
&SET DICUSERDATA &E(TVAR)
```

DIERRFILTER This variable can be used to control which errors are actually meaningful to you at a point in time during a translation. A description of the error filter can be found in the *DataInterchange Programmer's Reference* under the section "Error Filtering" and under the description for "ERRFILTER."

DIEXPTRACE This variable, when given a nonzero value (&SET DIEXPTRACE 1), causes DataInterchange to create a TRACE of the results of all expression evaluations. When tracing is active, DataInterchange will write out message TR0411 to the PRTFILE for each expression. The message will show the expression being evaluated and the result of the evaluation. Tracing will remain active until the DIEXPTRACE is given a zero value (&SET DIEXPTRACE 0).

Note: The TR0410 and TR0411 messages always occur as the first messages for a transaction. They are not merged with any other error messages for the transaction.

DIMAPCHAIN This variable can be used when an inbound transaction is required by more than one application program. It allows more than one mapping to be executed for the specified transaction. The last value given to DIMAPCHAIN in a mapping will establish the application sender ID value that will be used to locate the next mapping to execute. For example, if MAPABC had this coded:

```
&SET DIMAPCHAIN APPLICATIONB
```

the inbound transaction would be translated using map MAPABC, and then it would be translated using the map that is associated with application sender ID APPLICATIONB. The DIMAPCHAIN command will cause all maps indicated by each DIMAPCHAIN command to be translated, whereas the DIMAPSWITCH command will stop translating the map that has the DIMAPSWITCH variable in it, and literally switch to the new map indicated in the command.

DIMAPSWITCH This variable can be used when data being received needs to be inspected before it can be determined exactly what mapping should be done against the transaction. It allows you to switch the map that is being executed dynamically based on the data that is being received. A map could be created to initially look at the data being received. Only those data elements necessary to make a mapping decision would be mapped. DataInterchange would determine the real map to be used by interpreting values resulting from conditional logic expression. For example, a map would contain conditional logic expression:

```
&IF(X > Y) &SET DIMAPSWITCH APPLICATIONA
```

Here, if X is greater than Y, the mapping identified with an application sender ID value of APPLICATIONA would be used to translate the transaction.

DISAPSEQ This variable can be used to allow saving of the SAP IDOC record sequence number on the first error encountered during outbound processing. The sequence number may be provided through the application or using the DataInterchange accumulators. Variable DISAPSEQ is captured in the SAP status record to indicate the first record in error. For more information, see the *DataInterchange Programmer's Reference*.

DIVALLEVEL This variable can be used to control the level of validation done. It can have the same values as the validation level specified in a usage record, which are: 0 (no validation), 1 (validation tables activated), and 2 (validation tables plus type checking). Any value other than 0, 1, or 2 will be treated as a 0.

Mapping Your Application Data to an EDI Standard Transaction Set

DIVALTYPE	This variable can be used to control the data types for which data type checking is done (validation level of 2). The types that may be specified are DT, TM, N, R, CH, AN, A, and HX. They must be specified in uppercase and separated by a comma. Any value specified that is not in the list above will be ignored. For example, to activate DT, TM and HX validation, the following could be done: <code>&SET DIVALTYPE DT, TM, HX</code>
DIVARTRACE	This variable, when given a nonzero value (<code>&SET DIVARTRACE 1</code>), causes DataInterchange to create a TRACE of all accesses to variables. When tracing is active, DataInterchange will write out message TR0410 to the PRTFILE for each variable access. The message will indicate the variable being accessed and its current value. Tracing will remain active until the DIVARTRACE is given a zero value (<code>&SET DIVARTRACE 0</code>).

Mapping Techniques for Literal Keywords

As you will see in the following examples, deciding where to save variables, execute expressions, and subsequently use variables is fundamental. To determine where these operations should be done, it is essential to understand the order in which the translator executes element mapping instructions. As stated earlier, transactions are always processed starting with the first data element of the first segment and proceeding to the next element of the same segment, after which the next segment is processed, and so on. If one element has repeat mappings, the instructions are executed in a top-down fashion. It is equally important to understand the difference between a mapping that may contribute to the output versus one that does not. For example, specifying **&SAVE variable** or **&SET variable** will not contribute to the output directly. Only direct mappings such as specifying an Application Field Name, forcing a literal value, specifying **&USE variable**, or evaluating an **&IF** expression, may contribute to the output. Determining where to save and use variables is different for inbound and outbound translation.

For send processing, proper placement is easy for mappings that contribute to the output because you know which element requires the result. Proper placement for a mapping that does not contribute is not as obvious because there may be no data element relationship with this action. In most cases, you will save an application value into a variable, check or manipulate the variable, then use it. In other cases, it may not be appropriate to perform all these actions in successive repeat element mappings. For instance, you may have two independent looping structures (records) and you need to save a value from a particular iteration in the first loop. This particular value must be saved while the translator is processing this first loop. The saved variable can then be inspected and manipulated in the second repeating loop to provide the desired result. In summary, the location of a mapping that saves a variable can be far from a corresponding mapping that actually uses the variable. The best technique for deciding where to save values is to do so at or near processing points in the map when the translator is working on the corresponding record.

Note: Typically, the map for saving a value into a variable will precede the event of actually using the variable. A feature called **&DEFERRED &USE** can be utilized in the event that output is necessary in an earlier segment than where the final result will be set.

For receive processing, the situation is reversed. It is easy to decide where to save variables because you already know which element value in the input you need to work with. Proper placement for a mapping that contributes to your output application data is not as obvious because there may be no data element relationship with this action. In most cases, you will save a data element value into a variable, check or manipulate the variable, then use it in your application structure. In other cases, it may not be appropriate to perform all these actions in successive repeat element mappings. For instance, you may need to save values from two different segments, compare them, and write the result to an application field. One input element that needs to be saved is in the header section of the EDI transaction and the second is in an outer loop (the name loop for example). The output application field that needs to be

output after comparing these two variables is related to the detail loop (line item loop for example). It is most appropriate to use the result within the detail loop because this may have an independent repeating record that is not associated with the name or header segments. Hence, the comparison and use of the result should be done within this detail loop that controls the creation of the detail record. The best technique for deciding where to use variables is to do so at or near processing points in the map when the translator is working on the corresponding record.

Note: For receive processing, the map for saving a value into a variable must precede the event of actually inspecting or using the variable. The feature of **&DEFERRED &USE** cannot be used during inbound translation.

Examples of Using Literal Keywords and Named Variables

Example 1: For an inbound transaction, if your trading partner sends you the city name, state abbreviation, and zip code in three fields, but your database puts all of this information in one 30-byte field, ADDRESS, you could use **&SAVE** to put the address information into the single field:

1. Map the city name data element using the literal **&SAVE citystzip,1,19**.

This creates the named variable **citystzip** and places the city standard data received in the first 19 bytes of the named variable.

2. Map the state abbreviation data element using the literal **&SAVE citystzip,20,2**.

This places the state abbreviation standard data received in the bytes 20 and 21 of the named variable.

3. Map the zip code data element using the literal **&SAVE citystzip,22,9**.

This places the zip code standard data received in bytes 22 through 30 of the named variable.

4. Repeat the mapping of the zip code data element. Specify application field ADDRESS and the literal **&USE citystzip**.

DataInterchange uses the value in the named variable that was concatenated in the first three steps.

Example 2: For an outbound transaction, you want to provide a telephone number, either of a specific contact (CONTACTPHONE) or of the organization (ORGPHONE). If a contact phone number is provided, you want to use it; otherwise, you want to use the organization phone number. CONTACTPHONE occurs before ORGPHONE.

1. In the first mapping of the phone number data element, specify application field CONTACTPHONE and the literal **&IFDATA &SAVE Tphone**.

This creates the named variable **Tphone** only if the CONTACTPHONE field contains data.

2. Repeat the mapping of the phone number data element. Specify application field ORGPHONE and the literal **&IFNOVAR &SAVE Tphone**.

This creates the named variable **Tphone** only if the named variable did not already exist (CONTACTPHONE did not contain any data).

3. Repeat the mapping of the phone number data element. Specify the literal **&USE Tphone**.

Example 3: For an inbound transaction, the application data field NAME should receive the value from standard data element 123 (CUSTNAME) in the first occurrence of the name and address loop, or from the data element (ORGNAME) in the second occurrence of the loop, if the first occurrence does not contain data.

Mapping Your Application Data to an EDI Standard Transaction Set

1. When mapping the first occurrence of the loop, specify the literal **&IFDATA &SAVE Name**. Do not specify an application field.

This will create the named variable Name only if the first occurrence contains data.

2. When mapping the second occurrence, specify the literal **&IF(Name EQ ' ') &LSAVE Name**. Do not specify an application field.

This will create the named variable Name only if it does not already exist (the first occurrence did not contain any data). &LSAVE is used so that the value of Name from the first occurrence of the loop is not disturbed.

3. Repeat the mapping for the second occurrence. Specify application field NAME and the literal **&USE Name**.

This will use either the value saved in step 2 (first occurrence did not contain a value) or the value saved in step 1 (first occurrence did contain a value). All future occurrences of the loop and unrelated loops and/or segments will only see the value of **Name** saved in step 1.

Example 4: For an inbound transaction, the application data field NAME should receive the value from standard data element 123 (CUSTNAME) in the second occurrence of the name and address loop, or from the data element (ORNAME) in the first occurrence of the loop, if the second occurrence does not contain data.

1. When mapping the first occurrence of the loop, specify the literal **&SAVE Name**. Do not specify an application field.

This will create the named variable Name.

2. When mapping the second occurrence, specify the literal **&IFDATA &SAVE Name**. Do not specify an application field.

This will overlay the named variable Name only if the second occurrence contains data.

3. Repeat the mapping for the second occurrence. Specify application field NAME and the literal **&USE Name**.

This will use either the value saved in step 2 (second occurrence contained a value) or the value saved in step 1 (second occurrence did not contain a value).

Example 5a: For an outbound transaction, either application data field ORDQTY or MINQTY should supply a value for a standard data element. The field with the largest value should be used. If neither ORDQTY or MINQTY contain data then a value of 100 should be used.

1. In the first mapping of the data element, specify application field ORDQTY and the literal **&SAVE QTY1**.

This saves the value of the application field to a named variable. If the application field does not contain a value, the variable value is zero.

2. Repeat the mapping of the data element. Specify application field MINQTY and the literal **&SAVE QTY2**.

This saves the value of the application field to a named variable. If the application field does not contain a value, the variable value is zero.

3. Repeat the mapping of the data element. Specify the literal **&IF(QTY1 >= QTY2) &USE QTY1 100**. Do not specify an application field.

This compares the values of the named variables. If QTY1 is greater than or equal to QTY2, DataInterchange will use QTY1. If neither ORDQTY or MINQTY contain a value then both QTY1 and QTY2 will have a value of 0 and will therefore be equal. However, a value of 0 is not significant, which causes DataInterchange to use the default literal value of 100.

4. Repeat the mapping of data element 123. Specify the literal **&IF(QTY1 < QTY2) &USE QTY2**. Do not specify an application field.

This compares the values of the named variables. If QTY1 is less than QTY2, DataInterchange will use QTY2.

Example 5b: Assume you have created a mapping similar to that of example 5a, but forgot to do step 1. Because it is necessary to define QTY1 before using it, you need to insert a mapping of this data element, making it the first occurrence of the loop. Instead of remapping all of the occurrences, you can use the insert function from panel TP13 to insert a new data element mapping.

1. From panel TP13, place an **i** next to the occurrence you want to insert above.
2. Specify application field ORDQTY and the literal **&SAVE QTY1**, and press Enter.

Example 6: Your application can generate three different discount rates: the regular discount (REGDISC), a volume discount (VOLDISC), and a special discount (SPECDISC). For an outbound transaction, if application data field SPECDISC contains a value, you want to use it for standard data element 456. However, if SPECDISC does not contain a value, then the larger value of either REGDISC or VOLDISC should be used.

1. In the first mapping of data element 456, specify application field REGDISC and the literal **&SAVE REGDISC**.

This saves the value of the application field to a named variable. If the application field does not contain a value, the variable value is zero.

2. Repeat the mapping of data element 456. Specify application field VOLDISC and the literal **&SAVE VOLDISC**.

This saves the value of the application field to a named variable. If the application field does not contain a value, the variable value is zero.

3. Repeat the mapping of data element 456. Specify application field SPECDISC and the literal **&IF(REGDISC >= VOLDISC) &E(REGDISC)**.

This mapping will be executed only if the value of REGDISC is greater or equal to the value of VOLDISC. If this is the case, then SPECDISC will be mapped to the data element. However, if SPECDISC does not contain any data, then the default literal value of &E(REGDISC) will be used.

4. Repeat the mapping of data element 456. Specify application field SPECDISC and the literal **&IF(REGDISC < VOLDISC) &E(VOLDISC)**.

This mapping will be executed only if the value of REGDISC is less than the value of VOLDISC. If this is the case, then SPECDISC will be mapped to the data element. However, if SPECDISC does not contain any data, then the default literal value of &E(VOLDISC) will be used.

Notes on Examples 5 and 6: Examples 5 and 6 illustrate the differences between &IF, &USE, and &E:

- &IF is used to determine if a mapping should be executed. The value of the &IF expression is not used in the mapping; it only controls the execution of the mapping. If the expression is true (nonzero value), the mapping is executed. If the expression is false (zero value) the expression is not executed. In examples 5 and 6, only one of the maps in steps 3 and 4 will be executed because the expressions are mutually exclusive.
- &USE indicates that a named variable should be used as the primary source for data in the mapping. An application field cannot be specified, but you can have a default literal value if the variable name being used does not contain any data.
- &E is used exactly the same as a literal value, but instead of having a constant literal value, &E allows a literal value to be computed, taken, or computed and taken from a named variable. If the last

Mapping Your Application Data to an EDI Standard Transaction Set

operator of an expression is a Boolean or comparison operator, then the value of the expression will either be true (1) or false (0), and you can map these values.

For example, if a data element should contain a 1 if FLD1 is greater than FLD2, or a 0 if it is not, then mapping the expression `&E(FLD1 > FLD2)` would result in either a 1 or 0 being moved to the standard data element. If a data element should contain a 1 if FLD1 is greater than FLD2, or nothing if it is not, then the conditional mapping `&IF(FLD1 > FLD2) 1` would map the constant value of 1 only when FLD1 is greater than FLD2.

Example 7: For an outbound transaction, data element 321 should have a value of S if the value of application field SIZE is 6, 7, or 8; a value of M if SIZE is 9, 10, 11, or 12; and a value of L if SIZE is 13, 14, or 15. These values are specified in the translation table SIZETAB. However, for this transaction, SIZE can also be less than 6, in which case data element 321 should have a value of XS, or greater than 15, in which case data element 321 should have a value of XL.

1. In the first mapping of data element 321, specify application field SIZE and the literal **&SAVE Size**.

This saves the value of the application field to a named variable. If the application field does not contain a value, the variable value is zero.

2. Repeat the mapping of data element 321. Specify the literal **&IF(Size > 0 AND Size < 6) XS**.

This compares the value of the named variable to 6 and 0. If the value is less than 6 but greater than 0, DataInterchange uses the value XS.

3. Repeat the mapping of data element 321. Specify the literal **&IF(Size > 15) XL**.

This compares the value of the named variable to 15. If the value is greater than 15, DataInterchange uses the value XL.

4. Repeat the mapping of data element 321. Specify application field SIZE and the literal **&IF(Size >= 6 AND Size <= 15) &E(Size TS 'SIZETAB')**.

This compares the value of the named variable to 6 and 15. If the value is greater than or equal to 6, and less than or equal to 15, DataInterchange uses the translation table SIZETAB to determine the corresponding value for the named variable.

Example 8: Assume you have a lumber supply business and your trading partners are home builders in the area. The unit of measure on their orders ranges from inches to rods (one rod equals 5½ yards). You have implemented just-in-time inventory processes so if you receive an order that requires additional inventory, your order is immediately sent to your supplier. Your application stores all measurements in board feet, and therefore must convert all incoming data to board feet.

Your application fields are QUANTITY and UNITMEAS. Using the conditional processing literals, you would:

1. Map the unit of measure data element, using the literal **&SAVE UOM**. Do not specify an application field.

This saves the present value of the standard data element in the named variable UOM.

2. Repeat the mapping of the unit of measure data element, using the literal **&IF(UOM EQ 'IN' OR UOM EQ 'BF' OR UOM EQ 'RD') &SET UOMOK 1**. Do not specify an application field.

This statement enables you to determine if you have only the values you want, and if the statement is true, DataInterchange puts a 1 in the named variable UOMOK for later use.

3. Repeat the mapping of the unit of measure data element. Specify application field UNITMEAS and the literal **&IF (UOMOK EQ 1) &FORCE BF**.

This forces the value 'BF' into the application field only when the unit of measure is a valid value. The steps that follow will convert the data received into board feet.

4. Map the quantity data element, using the literal **&SAVE QTY**. Do not specify an application field.

This saves the present value of the standard data element in the named variable QTY.

5. Repeat the mapping of the quantity data element. Specify application field QUANTITY and the literal **&IF(UOM EQ 'IN') &FORCE &E(QTY/144)**.

This statement saves the value of quantity converted to board feet in the application field QUANTITY only if the incoming unit of measure was 'IN'.

6. Repeat the mapping of the quantity data element. Specify application field QUANTITY and the literal **&IF(UOM EQ 'RD') &FORCE &E(QTY*198/144)**.

This statement saves the value of quantity converted to board feet in the application field QUANTITY only if the incoming unit of measure was 'RD'.

7. Repeat the mapping of the quantity data element. Specify application field QUANTITY and the literal **&IF(UOM EQ 'BF') &FORCE &E(QTY)**.

This statement saves the value of quantity converted to board feet in the application field QUANTITY only if the incoming unit of measure was 'BF'.

8. Repeat the mapping of the unit of measure data element, using the literal **&ASSERT1(UOMOK EQ 1) &ERR(2,100,, 'Invalid unit of measure')**. Do not specify an application field.

This statement creates a translation error for anything that does not meet our criteria. Assume you received something in yards. You could add a repeat mapping for yards, then retranslate. The transaction would then pass through the translation.

An alternative to using conditional processing literals would be a single mapping using a translation table. For example, you could set up a translation table UOMDIV with the following entries:

Local	Standard
IN	144
RD	1.375
BF	1

If a translation table is used, then the mapping could be reduced to the following:

1. Map the unit of measure data element, using the literal **&SAVE UOM**.
2. Repeat the mapping of the unit of measure field, using the literal **&SET divisor &E(UOM TS 'UOMDIV')**.
3. Repeat the mapping of the unit of measure field, using the literal **&IF divisor NE 0) &FORCE BF**.
4. Map the quantity data element using the literal **&SAVE QTY**
5. Repeat the mapping of the quantity data element, using the literal **&FORCE &E(QTY/divisor)**
6. Repeat the mapping of the unit of measure data element, using the literal **&ASSERT1(divisor NE 0) &ERR(2,100,, 'Invalid unit of measure')**

This method might be preferred if the values of UOM are expected to change. If this happens, only the UOMDIV table needs to be updated rather than changing the mapping.

Example 9: For an outbound transaction, the application field TEST1 is an N2 data type and contains the value 100. You need to build a variable that contains "XX" in the first and second positions, "01" in the third, fourth, fifth, and sixth positions, and the value in the application field TEST1 beginning in the seventh position.

Mapping Your Application Data to an EDI Standard Transaction Set

To put this information into a single variable, do the following:

1. Map a data element using the literal **&SET TVAR**.

This creates the named variable TVAR and sets the variable to null.

2. Repeat the data element and map the data element using the literal **&SET TVAR,*,2 XX**.

This places "XX" in bytes 1 and 2 of the variable TVAR.

3. Repeat the data element and map the data element using the literal **&SET TVAR,*,4 01**.

This places "01" in bytes 3 through 6 of the variable TVAR.

4. Repeat the data element and map the data element using the application field TEST1 and a literal **&SAVE TVAR,*,7**.

This places "100" in bytes 7 through 9 of the variable TVAR. The variable TVAR now contains the following:

XX01 100

Note: At this point, the variable TVAR has been normalized to the data type of the application field TEST1, which is defined as an N2 data type.

To map TVAR position 3 through 6 to a standard data element with an ID data type, you would need to do the following because of the normalization of the variable to an N2 data type:

1. Map a data element using the literal **&E(CHAR(TVAR) SC 3.4)**.

The &E allows the literal to be computed. The CHAR forces the variable TVAR to have a data type of character for this instruction. The SC is a scaling function but can be used to substring character data. The resulting value would be "01".

An alternative to using the scaling function is as follows:

1. Map a data element using the literal **&E(CHAR(TVAR) TO 'CDEF')**.

The TO 'CDEF' is used to create the value by matching the default pattern ('ABCDEFGHIJK...') with the 'CDEF' pattern in this expression. The resulting value would be "01".

2. Map a data element using the literal **&USE TVAR,3,4**.

The resulting value would be "0.01".

To map TVAR position 7 through 9 to a standard data element with an R data type, you can move the data.

1. Map a data element using the literal **&USE TVAR,7,3**. The resulting value would be "1.00".

Example 10: For an outbound transaction, the application field TEST1 is an N2 data type and contains the value 123. Application field TEST2 is an N0 data type and contains the value 100.

To put this information into a single variable do the following:

1. Map a data element using the literal **&SET TVAR**.

This creates the named variable TVAR and sets the variable to null.

2. Repeat the data element and map the data element using the application field TEST1 and a literal **&SAVE TVAR,*,3**.

This places "123" in bytes 1 through 3 of the variable TVAR.

3. Repeat the data element and map the data element using the application field TEST2 and a literal **&SAVE TVAR,*,3**.

This places "100" in bytes 4 through 6 of the variable TVAR.

The variable TVAR now contains the following:

123100

Note: At this point the variable TVAR has been normalized to the data type of the application field TEST1 which is defined as an N0 data type.

To map TVAR position 1 through 3 to a standard data element with an R data type, you would need to do the following because of the normalization of the variable to an N2 data type:

1. Map a data element using the literal **&SET TEMPVAR &E(CHAR(TVAR) SC 1.3)**.

The &E allows the literal to be computed. The CHAR forces the variable TVAR to have a data type of character for this instruction. The SC is a scaling function but can be used to substring character data. The resulting value would be "123".

2. Repeat the data element and map the data element using the literal **&E(TEMPVAR / 100)**.

The resulting value would be 1.23.

An alternative is as follows:

1. Map a data element using the literal **&USE TVAR,1,3**.

The resulting value would be 123.

Control Data Literals

Audit and control are generally high priority items. If you have the need to add the internal trading partner ID or the nickname of your trading partner to your application data for each transaction, use the &TPID or &TPNICKN literals. If you need to include the application data format ID associated with the current transaction, use the &FORMAT literal. Finally, if you require the application control value associated with the current transaction, use the &ACFIELD literal. This value will only be correct after ALL the fields that comprise the application control field have been processed.

Mapping Specific Service Segment Fields (Receive Only)

Table 9-2 lists the literals that are provided so that every field within every service segment can be accessed using a combination of the segment ID (ISA, UNB, STX, and so forth) concatenated with a 2-byte number indicating the field within the segment wanted. The names created with this concatenation match the names of the fields defined in the E, I, T, U, and X profiles.

Invalid names (for example, ISA44) are not flagged as errors, but return no data. Using names that do not match the envelope type being received (for example, using ISA01 when EDIFACT service segments (UNB) are being used) is not an error, but no data is returned.

Table 9-2 (Page 1 of 2). Service Segments

Literal	nn Value	Segment
&ISA nn	01 through 16	ISA
&GS nn	01 through 08	GS
&ST nn	01 through 02	ST
&SE nn	01 through 02	SE

Mapping Your Application Data to an EDI Standard Transaction Set

Table 9-2 (Page 2 of 2). Service Segments

Literal	nn Value	Segment
&GE ⁿⁿ	01 through 02	GE
&IEA ⁿⁿ	01 through 02	IEA
&UNB ⁿⁿ	01 through 18	UNB
&UNG ⁿⁿ	01 through 13	UNG
&UNH ⁿⁿ	01 through 09	UNH
&UNT ⁿⁿ	01 through 02	UNT
&UNE ⁿⁿ	01 through 02	UNE
&UNZ ⁿⁿ	01 through 02	UNZ
&STX ⁿⁿ	01 through 12	STX
&BAT ⁿⁿ	01 through 01	BAT
&MHD ⁿⁿ	01 through 06	MHD
&MTR ⁿⁿ	01 through 01	MTR
&EOB ⁿⁿ	01 through 01	EOB
&END ⁿⁿ	01 through 01	END
&BG ⁿⁿ	01 through 07	BG
&EG ⁿⁿ	01 through 04	EG
&ICS ⁿⁿ	01 through 10	ICS
&ICE ⁿⁿ	01 through 02	ICE

Mapping Generic Service Segment Fields (Receive Only)

You can map received envelope data to application fields by using substitution keywords in the **Literal** field. The keywords indicate which service segment field is mapped to the application field.

Table 9-3 describes the substitution keywords you can use to map service segment fields. The Envelope Data Type column indicates the required data type for the service segment field. The Standard Data Type indicates the data type that DataInterchange uses for conversions from a standard data type to the application data type.

Table 9-3 (Page 1 of 2). Keywords for Mapping Envelope Data (Receive Only)

Keyword	Envelope Data Type	Standard Data Type	Envelope Data Mapped to Application
&I		A	Entire interchange service segment, up to the length of the application field
&ICN	CN or IV	AN	Interchange control number
&IIS	IS, AS, or RS	A	Interchange sender ID
&IIR	IR, AR, or RR	A	Interchange receiver ID
&IDT	DT	DT	Interchange date
&ITM	TM	TM	Interchange time
&IPW	PW	A	Interchange password
&IAP	AP	A	Interchange application reference

Mapping Your Application Data to an EDI Standard Transaction Set

Table 9-3 (Page 2 of 2). Keywords for Mapping Envelope Data (Receive Only)

Keyword	Envelope Data Type	Standard Data Type	Envelope Data Mapped to Application
&IVR	VR	A	Interchange version/release
&IGT		N0	Interchange total number of groups
&ICT	CT	N0	Interchange control total from the interchange trailer segment
&ITT		N0	Interchange total number of transactions
&G		A	Entire group service segment, up to the length of the application field
&GCN	CN or IV	AN	Group control number
&GFG	FG	A	Functional group ID
&GAS	AS, IS, or RS	A	Group application sender ID
&GAR	AR, IR, or RR	A	Group application receiver ID
&GDT	DT	DT	Group date
>M	TM	TM	Group time
&GPW	PW	A	Group password
&GVR	VR	A	Group version
&GLV	LV	A	Group release
>T		N0	Group total number of transactions
&T		A	Entire transaction service segment, up to the length of the application field
&TCN	CN or IV	AN	Transaction control number. The &TCN keyword is valid for both send and receive. See "Mapping Service Segment Fields (Send Only)" on page 9-36 for an explanation on the use of &TCN for application assigned control numbers.
&TTC	TC	A	Transaction code
&TVR	VR	A	Transaction version
&TLV	LV	A	Transaction release
&TTS		N0	Transaction total number of segments

DataInterchange interprets any literal beginning with an ampersand (&) as a special keyword. To use a literal that begins with an ampersand, use two ampersands. The translator discards the first one and uses the remaining characters as literal data. For example, if you enter &T in the **Literal** field, the translator moves the entire transaction service segment to the application data. If you enter &&T in the **Literal** field, the translator removes the first & and uses &T as literal data. The service segments (ISA, GS, ST, UNB, UNG, and so on) are not provided in the list of segments that can be mapped for trading partner transactions. Because they are not provided in the list, a direct mapping of a field from a service segment is not possible. In order to use one of the literals from Table 9-2 on page 9-33 or Table 9-3 on page 9-34, you have to map, or repeat map, some other data element defined in the transaction. When one of the special literals is used, DataInterchange knows that the value of the data element being mapped should be ignored, and the value of the special literal should be used instead.

Restrict the mapping of service segment fields to data elements in nonrepeating segments. Select a data element in the first nonrepeating segment that you know will always be present in the received transaction data. The segment must be present in the data being received for the mapping instructions to be

Mapping Your Application Data to an EDI Standard Transaction Set

executed. Repeat the element mapping as many times as is necessary to map (or &SAVE) all of the service segment fields you need. DataInterchange recognizes the substitution keywords and moves the value from the associated service segment field rather than the transaction data element that is currently being mapped to the application field or named variable. Data conversions, table translations, and user exits are possible when service segment special literals are used.

Mapping Service Segment Fields (Send Only)

During the send mapping process, the &TCN special literal may be used to indicate the application field which contains the message or transaction control number. Any field from a segment currently mapped may be chosen and either mapped or repeated, and the special literal &TCN used to identify the application field containing the control number.

- If the application field is part of a structure that occurs more than once, the first structure is the only one that will be used.
- It is possible to have more than one mapping which contributes to the transaction control number. The data from each mapping will be concatenated with the current data from previous mappings.
- If &DATE is used in the mapping, the format will be *yyyymmdd*. If &TIME is used in the mapping, the format will be *hhmmss*.

The transaction control number is extracted at translate time so if delayed enveloping is used and &DATE or &TIME, or both, are used to construct the control number, the date and time will be the date and time of translation and not the date and time of enveloping.

- Translation tables, validation tables, and user exits can be used during &TCN mappings the same as they can be used in any other mapping.
- The transaction control number generated by the application will be truncated to the maximum length for a control number allowed by the standard, never to exceed fourteen bytes.
- It is possible to combine delayed enveloping with application assignment of control numbers. If this is done, when an envelope operation is requested, the transactions will be sorted such that all transactions for which DataInterchange will assign control numbers will occur before the transactions for which the application has assigned control numbers. Also, during an envelope operation, a switch from a transaction that requires DataInterchange to assign the control number to a transaction with application assigned control numbers will cause a new interchange to be started. Transactions with application assigned control numbers will be sorted by the value of the transaction control number.

Error message TR0115 will be displayed if the application field containing the control number was not provided, if it contained all blanks, or was otherwise invalid.

Error message TR0116 will be issued if the control number assigned is a duplicate within the group/interchange. The translator requires that message control numbers must be unique within the group. If groups are not being used, then message control numbers must be unique within the interchange.

These errors are considered level 3 errors.

You must fix the application so that the message control numbers are unique within the interchange or group.

Validation During Mapping

The following list describes the type of validation used during translation, based on the type of data being validated.

Data Type Validation

A	The ALPHANUM validation table shipped with DataInterchange, with numeric digits (0 through 9) removed, is used to validate the data.
AN	The ALPHANUM validation table shipped with DataInterchange is used to validate the data.
BIN	A generic data type that encompasses the P, L, Z, B, I and H data types.
CH	The CHARSET validation table shipped with DataInterchange is used to validate the data.
N	Only digits, leading or trailing sign characters, and leading or trailing blanks are allowed.
R	Only digits, decimal notation, exponential notation, leading or trailing sign characters, and leading or trailing blanks are allowed.
DT	Must be a valid date according to the format specified during the mapping process. (See the Mapping Tip on page 9-60).
TM	Must be a valid time.
-	Validation is done automatically because a data conversion is required.

Table 9-4 shows the type of validation done for every possible mapping between application data types (down) and standard data types (across) during a TRANSLATE TO STANDARD processing.

Table 9-4. Type of Validation During Translate to Standard

Data type	A	AN	BIN	N	R	DT	TM
A	A	R	AN	N	R	DT	TM
AN	A	R	AN	N	R	DT	TM
AC	A	R	AN	N	R	DT	TM
CH	CH	R	CH	N	R	DT	TM
DT	DT	DT	DT	DT	DT	DT	DT
TM	TM	TM	TM	TM	TM	TM	TM
R	R	R	R	R	R	DT	TM
N	N	N	N	N	N	DT	TM
P	-	-	-	-	-	DT	TM
L	-	-	-	-	-	DT	TM
Z	-	-	-	-	-	DT	TM
B	-	-	-	-	-	DT	TM
I	-	-	-	-	-	DT	TM
FN	CH	R	CH	CH	CH	CH	CH
H	-	-	-	-	-	DT	TM

Table 9-5 on page 9-38 shows the type of validation done for each possible mapping between application data types (down) and standard data types (across) during TRANSLATE TO APPLICATION processing. For a description of data types, see Table 6-3 on page 6-21 and Table 6-4 on page 6-33.

Mapping Your Application Data to an EDI Standard Transaction Set

Table 9-5. Type of Validation During Translate to Application

Data type	A	AN	N	R	DT	TM
A	A	A	N	R	DT	TM
AN	AN	AN	N	R	DT	TM
AC	AN	AN	N	R	DT	TM
CH	CH	CH	N	R	DT	TM
DT	DT	DT	DT	DT	DT	DT
TM	TM	TM	TM	TM	TM	TM
R	R	R	R	R	DT	TM
N	N	N	N	R	DT	TM
P	R	R	N	R	DT	TM
L	R	R	N	R	DT	TM
Z	R	R	N	R	DT	TM
B	R	R	N	R	DT	TM
I	R	R	N	R	DT	TM
FN	CH	CH	CH	CH	CH	CH
H	R	R	N	R	DT	TM

Mapping Guidelines

This section describes several mapping guidelines. These guidelines are recommendations and are not necessarily enforced by the mapping facility.

Defining a Data Hierarchy for Mapping

For occurrence mapping to be correct, the data format must present a hierarchy that defines the logical relationships among data structures. You can pass the related structures to the translator separately or together. Use the **Passed Separately?** field in the application data format to indicate whether or not a structure is passed separately or as part of a parent structure. Consider this example:

Structure hierarchy	Max use	Passed separately?
Structure 1	5	Y
Structure 2	2	N
Structure 3	2	Y

Structure 1 is the parent of structures 2 and 3. The relationship can be stated as:

For each occurrence of structure 1, there can be two occurrences of structures 2 and 3.

Structure 2 is passed with its parent structure. The application data must, therefore, provide space for both occurrences of structure 2 even though data for only one occurrence may be present. Because structure 3 is passed separately, the maximum use indicates the maximum number of occurrences that will be processed. The application data does not have to provide space for unused occurrences.

Mapping Nonrepeating Segments

You can map a nonrepeating segment that is not in a loop to almost any part of the data format. For receive mapping, you can map the data elements of a nonrepeating segment to any data field. For send mapping, the data elements can be mapped from any data field. If the field is in a repeating structure, however, the data is taken from the first occurrence only.

Mapping Repeating Segments Not in Loops

Single-occurrence mapping for these segments is the same as for nonrepeating segments that are not in a loop. See “Mapping Nonrepeating Segments.”

For multiple-occurrence mapping, you can map the data elements to or from any fields within the domain of the data structure associated with the segment. Consider this example:

Segments	Structures
Segment A	Structure 1
Segment B (5)	Structure 2
	Field a
	Field b
Segment C	Structure 3
	Field c

Structure 2 defines the domain for segment B, which has five occurrences. You can map any field in structure 2 to or from any data element in segment B. For send mapping, you can also map a nonrepeating field from outside the domain (such as field c above) to segment B. Each occurrence of the data element will contain the same data. For receive mapping, you can map a repeating data element to a nonrepeating field, but only the last occurrence of the data element is used.

Mapping Loops

Single Occurrences

For single occurrences, you can map any data element to or from any field within the current domain. If there is only one level of looping, the domain is the entire transaction. An outer loop sets the domain for inner loops. Here is an example:

Segments	Structures
Segment A	Structure 1
Segment B	Structure 2
Segment C	Structure 3
Segment D	
Segment E	

Outer loop A/E is associated with structure 1, which sets the domain for the loop. You can map single occurrences of data elements in loop A/E, which includes loop B/C, to or from any field in structure 1, which includes structures 2 and 3.

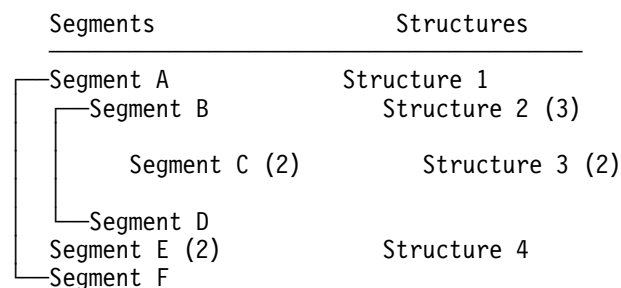
Multiple Occurrences

You can map multiple occurrences of data elements to or from any fields that are in the domain of the structure associated with the loop. In the previous example, you can map data elements in loop B/C to or from any fields in structure 2. For send mapping, you can also map a nonrepeating field from outside the domain to a data element in the repeating loop. Each occurrence of the data element will contain the same data. For receive mapping, you can map a data element from the repeating loop to a nonrepeating field, but only the last occurrence of the data element is used.

Repeating Segment Within a Loop

Single-occurrence mapping is the same as for a nonrepeating segment not in a loop, with this exception: the mapping should stay within the current domain.

For multiple occurrences, you can map data elements to or from any fields within the domain of the structure associated with the segment. The structure associated with the segment should be within the domain of the structure associated with the loop. Consider the next diagram, which shows repeating segments within loops.



In this example, you can map data elements in repeating segment C to or from any fields in structure 3. Structure 3 is within the domain of structure 2, which is associated with loop B/D. Segment E, on the other hand, should not be mapped to structure 3. Because both segments repeat and are not in the same domain, there is no certain way of determining which segment occurrence maps to which structure occurrence. The mapping will take place, but results are unpredictable.

For send mapping, you can map a nonrepeating field from outside the domain to the repeating segment. Each occurrence of the data element will contain the same data. For receive mapping, you can map a data element from the repeating segment to a nonrepeating field, but only the last occurrence of the data element is used.

Mapping Data Elements to and from Structure Fields

For send mapping, if you map a nonrepeating field to a data element in a repeating segment or loop, each occurrence of the segment or loop will contain data from the field.

For receive mapping, if you map a nonrepeating data element to a field in a repeating structure, each occurrence of the structure will contain data from the data element.

You can qualify a group of data elements using one data element in the group. You can then map the group in different ways depending on the qualifier value.

Send Mapping One to Many, Many to One

Using the **Repeat** action, you can map any field to as many data elements as you like. To map a concatenation of fields to one data element, use the **Special handling** option on the Map Data Element panel (TP10) and the concatenation fields on the Special Handling for Sending panel (TP30). See “Concatenating Data for a Data Element (Send Only)” on page 9-61 for more details.

Receive Mapping to Multiple Fields

Use the **Repeat** action to repeat a data element mapping for a different data format field. Use **Repeat** and the **Special handling** option on the Map Data Element panel (TP10) to map substrings of a data element to different application fields. The guidelines for mapping repeating segments and loops also apply to repeated data element mappings. See “Substringing Data for an Application Field (Receive Only)” on page 9-64 for more details.

Mapping Floating Segments

During the standards customization process, you can define a floating segment that can occur anywhere in the transaction mapping.

When creating transaction mappings, use the **Block** and **Copy** actions to place the floating segment exactly where you want it to be built, then map the segment like any other segment.

For receive transactions, a floating segment has meaning only when it follows the same segment in both the transaction data being received and the mapping. For example, if NTE is a floating segment and the mapping is:

```
BEG
CUR
NTE
PER
```

then the NTE segment is only processed if it follows the CUR segment in the transaction data. NTE segments following the BEG or PER segments are ignored, just like any other segment that is defined as part of the transaction but not mapped. This situation is true even if CUR is an optional segment that is not included in the transaction data. If the input transaction data is:

```
BEG*.....!
NTE*.....!
PER*.....!
```

the NTE segment is ignored because it does not follow the CUR segment.

A floating segment gets its meaning from the segment it follows. In this example, the NTE segment follows the BEG segment, but the mapping indicates an interest only in the NTE segments that follow the CUR segment.

Mapping floating segments requires additional work if the floating segments apply to a loop of segments, but appear at the end of the loop after optional segments. The floating segment must be copied and mapped after each of the optional segments to ensure that the floating segment is mapped and returned as application data.

Processing Bounded Loops

A bounded loop is a loop that is preceded by an LS (loop-start) segment and followed by an LE (loop-end) segment. Normally DataInterchange can determine the correct position within a transaction being received by the presence of mandatory segments defined for the transaction. However, in some transactions, the only way to determine the correct position within a transaction is to inspect the value of the LS01 field (Loop ID Code). The value for the LS01 field has changed over time, and DataInterchange has made no attempt to standardize this value. Instead, DataInterchange has the &LSID literal keyword.

The &LSID keyword can be used when mapping to provide the LS01 value that identifies the instance of the LS loop. You only need to use this keyword when the translator has no other way to determine which LS loop is provided, such as a mandatory segment between the two identical LS loops.

If you use the &LSID keyword during receive mapping, the translator inspects the contents of the LS segment, then does one of the following:

- If the content of LS01 matches any &LSID value, the translator uses that value to define the current position within the transaction definition from which translation will continue.
- If the value from the LS segment does not match any &LSID values, the translator assumes that the next LS segment for which an &LSID value was not provided is the proper position.

You can map the &LSID special literal to the LS segment or to any field in any segment that follows the LS segment. When the &LSID special literal is encountered during mapping, the translator assumes it applies to the current LS segment, if mapped to an LS segment, or to the LS segment immediately preceding the segment to which &LSID is mapped.

DataInterchange also provides two validations for bounded loops, and generates warning messages if the validations fail.

- Message TR0407 is generated if the LS01 value in the LS segment does not match the LE01 value in the LE segment.
- Message TR0408 is generated if there are LS segments that do not have corresponding LE segments.

Both TR0407 and TR0408 are only warning messages, so they do not generate return codes or have any effect on the translation of a transaction.

Preparing the Mapping Worksheet

To prepare a worksheet, start by printing a copy of your application data format and the standard you are mapping your data to. You might consider using the worksheet described in Chapter 7, “Defining and Working With Your Application Data,” and expanding it to accommodate your needs. Once your worksheet is prepared, compare the EDI data you expect to receive with the your application format. You would do the same for data you expect to send. Determine how the EDI standard segments and data elements match your records and fields. Using this information, complete your mapping worksheet for the transaction.

Figure 9-3 is an example of comparing received data with standard segments and data elements. Spaces have been inserted between data elements to make them easier to read.

Segment 1:

353 92 324 328 327 323 326 367 (Data element IDs)
BEG* 00* NE* P0123456* * * 910920* * C321! (EDI data)

Segments 2 and 3:

363 3
NTE* * Our store will be closed for v!
NTE* * acation from July 3 to July 14!

Segment 4:

350 330 355 212
P01* * 2* EA* 69.95!

Segments 5 and 6:

348 349 372
J2X* * * Glass fly rod!
J2X* * * 7 ft. # 6 line!

Segment 7:

350 330 355 212
P01* * 2* EA* 49.95!

Segments 8 and 9:

348 349 372
J2X* * * Single action!
J2X* * * fly reel!

Segment 10:

354 347
CTT* 2* 239.80!

Figure 9-3. Sample Purchase Order in EDI Format

Figure 9-4 on page 9-44 shows the record layout of the application file.

Mapping Your Application Data to an EDI Standard Transaction Set

Record Type/Field	Type	Length	Occurrences
Header			1
RECID	CH	4	
PONUM	CH	10	
PODATE	CH	8	
CONTRACT	CH	15	
Notes			3
RECID	CH	4	
NOTE	CH	30	
Line item			10
RECID	CH	4	
QTYORD	N0	6	
UM	CH	2	
PRICE	N2	8	
Description			2
RECID	CH	4	
DESCR	CH	15	
Totals			1
RECID	CH	4	
LINEITEMS	N0	2	
AMTDUE	N2	9	

Figure 9-4. Record Layout for Sample Purchase Order (Sending)

Figure 9-5 shows the worksheet that combines the information from the EDI data and the application file.

Segment/ Element ID	Structure/ Field Name	Type	Length	Max Use	Edits/ Notes
BEG	POHDR	ST		1	
324	PONUM	CH	10		
323	PODATE	CH	8		
367	CONTRACT	CH	15		
NTE	PONOTE	ST		3	
3	NOTE	CH	30		
PO1	POLINEITEM	ST		10	
330	QTYORD	N0	6		
355	UM	CH	2		
212	PRICE	N2	8		
J2X	PODESCR	ST		2	
372	DESCR	CH	15		
CTT	POTOTALS	ST		1	
354	LINEITEMS	N0	2		
347	AMTDUE	N2	9		

Figure 9-5. Mapping Worksheet for Sample Purchase Order (Receiving)

Now that you have prepared your worksheet, you are ready to begin mapping your data.

Creating a Trading Partner Transaction (Map)

The way you create a send transaction is very similar to the way you create a receive transaction. The following table describes the steps necessary to create a trading partner transaction.

Table 9-6. Steps for Creating a Trading Partner Transaction (Map)

Step	Action	See
1	Define the transaction by creating a new transaction or by copying and changing an existing transaction.	"Defining a Transaction" on page 9-47
2	Map the EDI segments and data elements used in the transaction.	"Mapping Segments and Data Elements" on page 9-52
3	Specify how the data elements are used in the segment:	
	• Are they mapped to an application field?	"Specifying an Application Field Name" on page 9-57
	• Do you need an accumulator?	"Specifying an Accumulator and Action" on page 9-57
	• Do you need a literal?	"Specifying a Literal" on page 9-58
	• Is special handling required?	"Special Handling" on page 9-59
4	Specify how loops are handled.	"Specifying Occurrences" on page 9-67
5	Identify which trading partners will use this map.	"Specifying Trading Partners" on page 9-78

The following sections use Figure 9-6 on page 9-46 and Figure 9-7 on page 9-47 as samples to demonstrate how tasks are accomplished.

Mapping Your Application Data to an EDI Standard Transaction Set

APP1	PURCORD00	Application Data Format	Date: 98/04/22 Time: 13:13:29
------	-----------	-------------------------	----------------------------------

Data format ID Sample Purchase Order Send
Description BASESTRUCTURE
Base structure name _____
Application file name _____
Application file type. ____ +
Record ID position _____
Record ID length ____
Record ID data type ____ +
Beginning structure name _____
Ending structure name _____
Trading partner ID field _____
Generic routing code field _____
Interchange sender fields . Qualifier _____ ID _____
Interchange receiver flds . Qualifier _____ ID _____
Application TP ID field _____
EDI TP field _____

Field/Structure Name	Type	Length/Max Use	Passed Separately?	Record ID
BASESTRUCTURE	ST	1	Y	
HEADER	ST	1	N	
PONUMBER	AC	12		
PODATE	DT	8		
TERMS	ST	1	N	
DISCPCT	R	2		
DISCDAYS	R	2		
NETDAYS	R	2		
FOB	CH	2		
NOTE	ST	2	N	
NOTETEXT	CH	35		
NAMEANDADDRESS	ST	3	N	
ORGANIZATIONID	ID	2		
NAME	ST	1	N	
ORGNAME1	CH	18		
ORGIDCODE	CH	12		
ADDRESS	ST	1	N	
ORGADDR1	CH	18		
ORGCITYSTATEZIP	ST	1	N	
ORGCITY	CH	11		
ORGSTATE	CH	2		
ORGZIP	R	5		
LINEITEMS	ST	25	N	
ITEMNUMBER	AN	13		
ITEMDESC	ST	1	N	
DESCRIPTION	CH	28		
QUANTITY	R	4		
UNITOFMEASURE	ID	2		
UNITPRICE	R2	10		
TRAILER	ST	1	N	
ITEMCOUNT	R	2		

Figure 9-6. Sample Purchase Order Application Data Format (Send)

Mapping Your Application Data to an EDI Standard Transaction Set

APP1	PURCORD10	Application Data Format	Date: 98/04/22 Time: 17:37:15
------	-----------	-------------------------	----------------------------------

Data format ID Sample Purchase Order Receive
 Description BASESTRUCTURE
 Base structure name _____
 Application file name _____
 Application file type ____ + _____
 Record ID position _____
 Record ID length _____
 Record ID data type ____ + _____
 Beginning structure name _____
 Ending structure name _____
 Trading partner ID field _____
 Generic routing code field _____
 Interchange sender fields Qualifier _____ ID _____
 Interchange receiver flds Qualifier _____ ID _____
 Application TP ID field _____
 EDI TP field _____

Field/Structure Name	Type	Length/Max Use	Passed Separately?	Record ID
BASESTRUCTURE	ST	1	Y	
HEADER	ST	1	N	
PONUMBER	AC	12		
PODATE	DT	8		
TERMS	ST	1	N	
DISCPCT	R	2		
DISCDAYS	R	2		
NETDAYS	R	2		
FOB	CH	2		
NOTE	ST	2	N	
NOTETEXT	CH	35		
BILLTONA	ST	1	N	
BILLTNAME	CH	18		
BILLTOIDCODE	CH	2		
BILLTOADDRLN1	CH	18		
BILLTOCITY	CH	11		
BILLTOSTATE	CH	2		
BILLTOZIP	CH	5		
SHIPTONA	ST	1	N	
SHIPTNAME	CH	18		
SHIPTOIDCODE	CH	2		
SHIPTOADDRLN1	CH	18		
SHIPTOCITY	CH	11		
SHIPTOSTATE	CH	2		
SHIPTOZIP	CH	5		
ORDERBYNA	ST	1	N	
ORDERBYNAME	CH	18		
ORDERBYIDCODE	CH	2		
ORDERBYADDRLN1	CH	18		
ORDERBYCITY	CH	11		
ORDERBYSTATE	CH	2		
ORDERBYZIP	CH	5		
LINEITEMS	ST	25	N	
ITEMNUMBER	AN	13		
ITEMDESC	ST	1	N	
DESCRIPTION	CH	28		
QUANTITY	R	4		
UNITOFMEASURE	ID	2		
UNITPRICE	R2	10		
TRAILER	ST	1	N	
ITEMCOUNT	R	2		

Figure 9-7. Sample Purchase Order Application Data Format (Receive)

Defining a Transaction

To start mapping a transaction to DataInterchange, you need to add a new transaction. When you add a transaction, you tell DataInterchange basic information about the trading partner transaction, such as the transaction ID, a description of the transaction, whether the transaction is a send or receive transaction, and so forth.

Mapping Your Application Data to an EDI Standard Transaction Set

To add a new transaction, follow these steps:



1. From the Administrator's Menu (MP01), select **Trading Partner Transactions**. The Transaction Mappings panel (TP01) is displayed. From this panel you can:

- Add a new transaction mapping
- Copy a transaction mapping and change it
- Delete a transaction mapping
- List maps by mapping ID
- Display a list of segments in a transaction mapping
- Migrate a map from one standard to another
- Print a transaction mapping
- Update a transaction mapping
- Generate the control string for a transaction mapping
- Display the trading partner usages for a transaction mapping
- Specify whether standard compliance checking is on all inbound data or mapped data only.

Note: The TYPE column specifies whether the trading partner transaction was previously transferred from the host to DataInterchange Client. If the value in the TYPE column is C, I, or P, the TPT was previously moved to DataInterchange Client. It is questionable whether the host TPT should be changed, because the most recent copy of the map source is on DataInterchange Client.

- C Indicates the host map was converted to a DataInterchange Client map database format. It also indicates the control string residing on the host was generated on DataInterchange Client.
- I Indicates the host map has been converted to a DataInterchange Client map database format, but the host control string was generated using the host map.
- P Indicates the host has only the map header; the complete map is stored in the DataInterchange Client Map database. The control string, if any, was generated on DataInterchange Client.

2. To add a new map, type **A** in the action column next to any item, and press Enter.

Add	Copy	Delete	List	Map	migrate	Print	Update	Generate	Where used
TP01			Transaction Mappings						1 to 8 of 8
A	Transaction ID	Send/Recv	Gen Reqd	Type	Description				
	INVOICERCVE	R	N		Invoice mapping for receive				
	INVOICERCVE2	R	Y	P	New Invoice mapping for receive				
-	INVOICESEND	S	N		Invoice mapping for send				
-	INVOICESEND2	S	Y		New Invoice mapping for send				
-	PURCORDRCVE	R	Y		Purchase order mapping for receive				
-	PURCORDSEND	S	N	C	Purchase order mapping for send				
	SAMPLEPORECV	S	Y		Sample Purchase Order Receive				
-	SAMPLEPOSEND	S	Y		Sample Purchase Order Send				

The Add Transaction Mapping panel (TP02) is displayed.

Use this panel to identify your transaction, the standard you are using, the data format ID you are mapping, and whether you are mapping a send transaction or a receive transaction. Application control field names and lengths are optional. Any application control entries on this panel override any field with the AC data type defined in your application data format.

Mapping Your Application Data to an EDI Standard Transaction Set

TP02		Add Transaction Mapping	
Transaction ID	sampleposend		
Transaction description . . .	Sample Purchase Order Send		
Standard ID	x12v2400		
Standard transaction ID . . .	850		
Data format ID	purcord00		
Standard compliance check . .	S		
Send or receive (S/R)	S	+	
Application Control Field Name Length			
1	ponumber		12
2	podate		8
3			
4			
5			
6			
7			
8			

For the sample transaction, the fields were completed as follows:

In this field:

Enter:

Transaction ID

The name of this transaction mapping. You can use any combination of characters in the ALPHANUM table, up to 16 characters. Embedded blanks are not allowed.

Transaction description

A brief description of the transaction mapping. You can use any character data. The maximum length of this field is 46 characters

Standard ID

The name of the EDI standard used for this transaction mapping. You can use any combination of characters in the ALPHANUM table, up to 8 characters. Embedded blanks are not allowed. The system will ensure the standard exists in the DataInterchange database.

Standard transaction ID

The transaction set within the EDI standard used for this transaction mapping. You can use any combination of characters in the ALPHANUM table, up to 8 characters. Embedded blanks are not allowed. The system will ensure the standard transaction set exists in the DataInterchange database.

Data format ID

The application data format name. You can use any combination of characters in the ALPHANUM table, up to 16 characters. Embedded blanks are not allowed. The system will ensure the application data format exists in your DataInterchange system database.

Mapping Your Application Data to an EDI Standard Transaction Set

In this field:

Standard compliance
check

Enter:

Specifies whether the translator should validate segments and data elements that have not been mapped, or only validate segments and data elements that have been mapped. Compliance checking only occurs on inbound standard data (Receive Maps).

With a value of Y, DataInterchange executes compliance check instructions in the control string for segments and data elements up to the last data element mapped.

With a value of N, DataInterchange checks mapped segments and data elements as usual; in addition, it adds instructions into the control string to check segments, loops, and repeating segments not mapped against the standard.

Data Elements not mapped will also have instructions for condition checking and mandatory fields. This option is the implementation of full compliance checking of standards.

Send or receive (S/R)

S for a send transaction. If you were mapping a receive transaction, you would type **R**.

In this field:

Application Control
Field Name

Enter:

The business application's key for an individual transaction. Leave these fields blank to use the contents of the application control (AC) field in your application data format definition as the application control value. If you want to override the value in your application data format definition, or if none exists, use these fields to create one. If the application control value comes from more than one field in the data format, use the **Length** field to indicate how many characters are taken from each field. The combined length cannot exceed 35 characters. The translator concatenates the data from these fields, in the order listed (1–8), and places the result in the appropriate data element.

These fields become a search key which is used to retrieve information from the Transaction Store and identify information in reports. Therefore, make a search key that is meaningful to members of your organization that are not familiar with EDI. For example, if the application control value is determined from a purchase order number and a department number prefix, list the fields containing these values.

The application control value, whether from the application data format or specified in this field, must be mapped to an EDI standard data element. The data that goes into or comes out of the EDI standard data element, not the application value, is used to create the control value. For example, if the application field is data type N2 with a value of 12345, and you map it to a type R2 data element, the control value is 123.45. If the application field contains the value January and you use a translation table to convert it to 01 as the standard value, the control value is 01.

The application field must be directly mapped to the EDI standard data element for the application control value to be set. If a value other than the EDI standard data element value is needed, there are two special keywords that can be used to set the application control value as desired:

1. The **&LIT** *value* is used to concatenate a 1 to 11 character literal *value* into the field. If a value is not specified, blanks are concatenated into the field.
2. The **&VAR** *variable name* is used to concatenate data from the named variable into the field.

These special keywords are not supported in Interactive Entry Facility (IEF).

Note: Literals specified with **&LIT** and variables specified with **&VAR** in the mapping are not available in the AC field until the end of the translation process and will not appear in the AC field data that is moved using the **&ACFIELD** keyword.

Mapping Your Application Data to an EDI Standard Transaction Set

In this field:

Length

Enter:

The number of characters from the application field used in concatenating the application control.

The combined length of the **Application Control Field Names** cannot exceed 35 characters.

The two forms of concatenation are based on the data type of the fields being concatenated.

Data Type

Concatenation

Bn, Hn, In, Ln, Pn, Zn

The translator converts the data to a character string with leading zeros and a negative sign if appropriate, then uses the length to concatenate the right-most characters. If you omit the length, the translator uses 10 characters. The length cannot exceed 10 for these data types.

All others

If you specify a length, the translator concatenates the left-most characters for that length. If you omit the length, it uses the length of the field from the data format.

For the &LIT special keyword, if you specify a length from 1 to 11, this length is used to concatenate the literal value. If you omit the length, it defaults to the length of the literal value.

For the &VAR special keyword, you should specify the number of characters to be concatenated from the named variable. If you omit the length, it defaults to 35.

3. Press Enter to save this information.

When you press Enter, the Transaction Segments panel (TP06) is displayed.

Mapping Segments and Data Elements

Once you have defined a new trading partner transaction, you need to select the EDI segments that map to your application data format, and define how the data elements are used within each segment. Some segments can be used repeatedly, for more information, see "Specifying Occurrences" on page 9-67.

To map segments and data elements, start from the Transaction Segments panel (TP06). For information about getting to the Transaction Segments panel (TP06), see "Defining a Transaction" on page 9-47.

The Transaction Segments panel (TP06) displays the segments included in this transaction set. The action bar of TP06 changes to include the action "Float delete." The actions are listed in alphabetical order.

1. To map a segment, type **M** in the action column next to each segment you want to map. For the sample send transaction, the sample BEG segment has been selected.

Mapping Your Application Data to an EDI Standard Transaction Set

Block	Copy	Delete mapping	Float delete	Line	Map	Qualify	Repeat	
TP06				Transaction Segments			1 to 8 of 66	
Transaction ID : SAMPLEPOSEND								
Standard ID : X12V2400								
Standard transaction ID . . : 850								
Mapped segments are highlighted.								
A	Ref	Segment	Seg/Loop	Req	Max	Loop	Loop	Description
	Num	ID	Mappings	Des	Use	ID	Repeat	
m	001	BEG		M	0001	0	0	Beginning Segment
-	002	NTE		F	0100	0	0	Note/Special Informatio
-	003	CUR		0	0001	0	0	Currency
-	004	REF		0	0012	0	0	Reference Numbers
-	005	PER		0	0003	0	0	Administrative Communic
-	006	TAX		0	0003	0	0	Sales Tax Reference
-	007	FOB		0	0001	0	0	F.O.B. Related Informat
-	008	N1		0	0001	000100	200	Name

From this panel, you can also:

Action	Description
Block	The segment is marked for a subsequent Copy command. This can only be used on lines with segments whose Req Des field is "F" (for floating). The Transaction Segments panel (TP06) is redisplayed. This action does not appear if the transaction displayed does not contain floating segments.
Copy	The floating segment marked by a Block command is copied to the line following the one on which the Copy command is entered and the Transaction Segments panel (TP06) is redisplayed. If the desired target of the copy is within a loop, the Map command must be used to display the segments for the desired mapping of the target loop and then the Copy command is used on the target segment on the Loop Segments panel. For example, if the target loop is the third level of looping, then the mappings and segments of the two higher level loops must be displayed in order to reach the display of the target loop. This action does not appear if the transaction displayed does not contain floating segments.
Delete mapping	<p>This is used to delete a segment or loop mapping definition. The Delete mapping command does not delete segments.</p> <ul style="list-style-type: none"> For segments (not within a loop), if only one mapping exists, that mapping is deleted and the Transaction Segments panel (TP06) is redisplayed with the mapping deleted. If more than one mapping exists, the Segment Mappings panel (TP11) is displayed to allow selection of the segment occurrence mapping to delete. For loops (entered on first segment of a loop), the Loop Mappings panel (TP12) is displayed (even if only one mapping) to allow the user to verify the delete. <p>To delete individual data element mappings, use the Map command to navigate through the panels to the desired data element selection panel (the Data Elements panel (TP09)) and use Delete mapping on that panel.</p>

Mapping Your Application Data to an EDI Standard Transaction Set

Action	Description
Float delete	<p>This command allows you to delete copied floating segments. This command removes the copied segment from the list of segments. If the segment is mapped with multiple occurrences, the maps must first be deleted, and then the copied floating segment can be Float deleted. This action does not appear if the transaction displayed does not contain floating segments.</p>
Line	<p>This command allows you to reset the top of the scrollable list.</p> <p>There are two methods for using Line:</p> <ul style="list-style-type: none">• Tab to the item you want placed at the top of the list, type L, and press Enter.• At the command prompt, type L and the Ref Num of the item you want at the top of the list. The ID does not have to be exact; the program finds the nearest match. <p>Note: When you enter the Line action in combination with other actions, by default the Line action is processed last.</p>
Map	<p>This command is used to define the mapping of a segment or loop.</p> <ul style="list-style-type: none">• For segments (not within a loop)<ul style="list-style-type: none">– If the segment has not been previously mapped, one of the following actions is taken:<ul style="list-style-type: none">- If the segment has a max use of 1, the Data Elements panel (TP09) is displayed to allow mapping of the elements.- If it has a max use greater than 1, the Qualify Segment Mappings panel (TP07) is displayed to allow qualification of the segment mapping and then the Data Elements panel (TP09) is displayed to allow mapping of the elements.- If the segment is qualified or mapped, it is marked as mapped but will not generate until it has mapped data elements.– If it has already been mapped, one of the following actions is taken:<ul style="list-style-type: none">- If it has more than one mapping, the Segment Mappings panel (TP11) is displayed to allow the user to select the segment occurrence mapping to work with and then the Data Elements panel (TP09) is displayed to allow mapping of the elements.- If it has only one mapping, the Data Elements panel (TP09) is displayed to allow mapping of the elements.

Action	Description
	<ul style="list-style-type: none"> • For loops (entered on first segment of the loop) <ul style="list-style-type: none"> – If the loop has not been previously mapped, the Qualify Loop Mapping panel (TP08) is displayed to allow qualification of the loop mapping and then the Loop Segments panel (TP14) is displayed to allow mapping of the loop segments. If you are mapping a hierarchical loop, the Hierarchical Loop Support panel (TP40) is displayed, offering optional hierarchical support before the Qualify Hierarchical Mapping for Send panel (TP41) or Qualify Hierarchical Mapping for Receive panel (TP42) is displayed. If the loop is qualified (not exit or cancel from TP08), then the first segment of the loop is marked as mapped, but the loop will not generate until you have mapped a data element in the first segment. – If it has already been mapped and it has only one mapping, the Loop Segments panel (TP14) is displayed to allow mapping of the loop segments. – If it has already been mapped and it has more than one mapping, the Loop Mappings panel (TP12) is displayed to allow the user to select the loop occurrence to work with and then the Loop Segments panel (TP14) is displayed to allow mapping of the loop segments.
Qualify	<p>Displays the qualification panel to allow changes. If it is a loop with only one mapping, the Qualify Loop Mapping panel (TP08) is displayed. If it is a loop with more than one mapping, the Loop Mappings panel (TP12) is displayed to allow selection of the occurrence to qualify. If it is a repeating segment with only one mapping, the Qualify Segment Mappings panel (TP07) is displayed. If it is a repeating segment with more than one mapping, the Segment Mappings panel (TP11) appears to allow selection of the occurrence to qualify.</p>
Repeat	<p>Creates another mapping occurrence of a mapped repeating segment or loop. The first occurrence of a segment or loop must be mapped before the repeat command can be used.</p> <ul style="list-style-type: none"> • For segments (not within a loop) <ul style="list-style-type: none"> – If entered on a nonrepeating segment, an error message is displayed. – If entered on a repeating segment, the Qualify Segment Mappings panel (TP07) is displayed to allow qualification of the segment mapping, then the Data Elements panel (TP09) is displayed to allow mapping of the elements for that occurrence. • For loops (entered on first segment of the loop) <ul style="list-style-type: none"> – If entered on a segment of the outermost loop, the Qualify Loop Mapping panel (TP08) is displayed to allow qualification of the loop mapping, then the Loop Segments panel (TP14) is displayed to allow mapping of the loop segments.

Commands are not allowed on any segments flagged with asterisks in the **Seg/loop Mappings** column. The asterisks indicate that the segments cannot be processed on this panel because they are part of a loop. To process the loop mapping, the loop segments must first be displayed by using **Map** to display the list of loop occurrences and using **Map** again on the occurrence to be mapped. The loop segments will then be displayed for mapping. For loops within loops, this process must be repeated until the desired loop level is reached.

Mapping Your Application Data to an EDI Standard Transaction Set

The Data Elements panel (TP09) will be displayed for each segment you mapped, listing the data elements in that segment.

2. Type **M** in the action column next to each data element you want to map. For example, the BEG segment of the sample send transaction would include data element ID 353, 92, 324, and 323.

Mapping Tip: Every mandatory data element should be mapped, or if you want to change one from mandatory to optional, consider customizing the standard to fit your needs. See Chapter 6, “Customizing EDI and Envelope Standards” for more information.

Delete mapping	Line	Map	Qualify	Repeat	Select		
<hr/>							
TP09			Data Elements		1 to 8 of 8		
Transaction ID . . . : SAMPLEPOSEND							
Loop ID :							
Segment ID : BEG							
Mapped data elements are highlighted. ***nnn points to a qualifying element.							
A	Seq	Element	Element	Req	Rel	Related Element	
	Num	ID	Mappings	Des	Type	Def	Description
m	1	353		M	ID		Transaction Set
m	2	92		M	ID		Purchase Order T
m	3	324		M	AN		Purchase Order N
-	4	328		O	AN		Release Number
-	5	327		O	AN		Change Order Sequence
m	6	323		M	DT		Purchase Order D
-	7	326		O	AN		Request Reference Num
-	8	367		O	AN		Contract Number

The Map Data Element panel (TP10) is displayed for each data element you selected. This panel is used to:

- Map a standard data element from an application data format field (see “Specifying an Application Field Name” on page 9-57)
- Control and/or map accumulators (see “Specifying an Accumulator and Action” on page 9-57)
- Specify a literal (see “Specifying a Literal” on page 9-58)
- Force data, based on logic (see “Specifying a Literal” on page 9-58)
- Request special handling (see “Special Handling” on page 9-59)

TP10	Map Data Element - BEG01
Transaction ID : SAMPLEPOSEND	
Loop ID . . . :	
Segment ID . . . : BEG	
Standard definition of data element	
Element ID . . . : 353	Elem qualifier :
Sequence num . . : 1	Description . . : Transaction Set Purchase
Type : ID	Req des : M
Minimum length : 2	Maximum length : 2
Definition of data element for this mapping transaction	
Application field name . .	+
Literal (enter on next line)	
Accumulator/action _ / _ _ / _ _ / _ _ / _	
Special handling (Y/N) . . N +	

Note: If you select a composite data element (the type is CD) on the Data Elements panel (TP09), then the Composite Data Element panel (TP60) is displayed. This panel looks and functions in a fashion similar to panel TP09.

Specifying an Application Field Name

When a data element maps to an application field name, specify the field name in the **Application field name** field of the Map Data Element panel (TP10).

Note: You can use the Prompt function to set the list of valid field names, or look at the Application Data Format. For more information, refer to the section “Prompt (F4)” on page 2-11.

TP10		Map Data Element - BEG03	
Transaction ID : SAMPLEPOSEND			
Loop ID . . . :			
Segment ID . . : BEG			
Standard definition of data element			
Element ID . . :	324	Elem qualifier :	
Sequence num . :	3	Description . :	Purchase Order Number
Type :	AN	Req des :	M
Minimum length :	1	Maximum length :	22
Definition of data element for this mapping transaction			
Application field name . .	ponumber +		
Literal (enter on next line)			
Accumulator/action _ / _ _ / _ _ / _ _ / _			
Special handling (Y/N) . . N +			

For example, the sample receive transaction would map element ID 324 of the BEG segment to PONUMBER.

Specifying an Accumulator and Action

When you want to keep running totals or accumulate numeric data, you can have DataInterchange store the information in an accumulator. For example, using the sample send transaction, keep a running total of the QUANTITY field in an accumulator.

On the Map Data Element panel (TP10), follow these steps:

1. In the **Application field name** field, type **QUANTITY**.
2. In the first *accumulator/action* field, type **T1** and **A**. These codes are described in “Using Accumulators” on page 9-3.
3. Press Enter.

Mapping Your Application Data to an EDI Standard Transaction Set

TP10	Map Data Element - P0102
Transaction ID : SAMPLEPOSEND	
Loop ID . . . :	000200 Multiple occurrence : LINEITEMS
Segment ID . . . :	P01
Standard definition of data element	
Element ID . . . :	330 Elem qualifier :
Sequence num . . :	2 Description . . : Quantity Ordered
Type :	R Req des : M
Minimum length :	1 Maximum length : 9
Definition of data element for this mapping transaction	
Application field name . . :	quantity +
Literal (enter on next line)	
Accumulator/action :	t1 / a _ / _ _ / _ _ / _
Special handling (Y/N) . . :	N +

Mapping Tip: For send transactions, accumulator actions will not be processed unless one of the following occurs:

- Data is generated for the standard field
- The accumulator is mapped

For receive transactions, accumulator actions will not be processed unless one of the following occurs:

- The data element associated with the accumulator is received
- The accumulator is mapped and at least the segment containing the data element is received.

Specifying a Literal

When your application data does not have specific information required by the standard, or when you and your trading partner agree on passing specified information, you can use a literal to specify what information is placed in a data element. For example, using the sample send transaction, assume that you and your trading partner agree that for the purchase order type, you will use the characters ZZ in the BEG segment. You would enter the information on the Map Data Element panel (TP10) as follows:

TP10	Map Data Element - BEG02
Transaction ID : SAMPLEPOSEND	
Loop ID . . . :	
Segment ID . . . :	BEG
Standard definition of data element	
Element ID . . . :	92 Elem qualifier :
Sequence num . . :	2 Description . . : Purchase Order Type
Type :	ID Req des : M
Minimum length :	2 Maximum length : 2
Definition of data element for this mapping transaction	
Application field name . . :	+
Literal (enter on next line)	
ZZ	
Accumulator/action :	_ / _ _ / _ _ / _
Special handling (Y/N) . . :	N +

Mapping Tip: Because the purchase order type is not part of the application data, the translator uses the value you supply. Remember that literals are case sensitive, and if you are defining one as a qualifier, it must be defined by the standard you are using.

In this case, this is the identifier for the purchase order type code. In the trading partner agreement, this was a mutually agreed upon value. Therefore, this element must contain the value **ZZ**. See “Using Literals” on page 9-4 for more information.

You can also use a literal when you want to supply a default value in case the application field does not contain a value or is not valid, or if you want to force a value of 0 into a standard data element. For example, using the send transaction, if the application field PODATE does not contain data, you want the system date to be put in the data element. You would complete the Map Data Element panel (TP10) as follows:

TP10 Map Data Element - BEG06

Transaction ID : SAMPLEPOSEND
 Loop ID . . . :
 Segment ID . . : BEG

Standard definition of data element
 Element ID . . : 323 Elem qualifier :
 Sequence num . : 6 Description . : Purchase Order Date
 Type : DT Req des . . . : M
 Minimum length : 6 Maximum length : 6

Definition of data element for this mapping transaction
 Application field name . . **podate** _____ +
 Literal (enter on next line)
&date _____

Accumulator/action / _ / _ / _ / _ / _
 Special handling (Y/N) . . **y** +

Mapping Tip: If your application does not always have a purchase order date, use the &DATE literal. This tells DataInterchange to use the system date whenever the application data does not provide a value. See “Using Literals” on page 9-4 for details.

Notice that the **Special handling** field specifies that this data element needs special handling.

Special Handling

The Special Handling for Sending panel (TP30) or the Special Handling for Receiving panel (TP31) is displayed when you request special handling from the Map Data Element panel (TP10). The Special Handling for Sending panel (TP30) lets you specify:

- Date edits
- Validation tables
- Translation tables
- User exits
- Concatenation position
- Concatenation length

The Special Handling for Receiving panel (TP31) lets you specify:

- Date edits
- Validation tables
- Translation tables
- User exits
- Substring positions

Mapping Your Application Data to an EDI Standard Transaction Set

- Substring lengths
- Segment and loop qualifiers

For example, the purchase order date from your application data is in the form *yyyymmdd*, but the standard expects it in the form *yymmdd*. Therefore, you have to do a date edit.

Note: User exits call a field exit routine to provide or process data. See the *DataInterchange Programmer's Reference* for more information about field exit routines.

Mapping Tip: Whenever your application requires a date in a format that is different from the standard, use the appropriate date edit code. For send transactions, all date formats are changed to *yymmdd*. For receive transactions, all date formats are changed from *yymmdd*. If the following predefined date edits do not satisfy your data edit requirements, you can use the any-to-any date conversion facility described in "Date Conversion Special Operators" on page 9-21. Valid date conversion values are:

Send Transaction		Receive Transaction	
Value	From	Value	To
1	ddmmyy	17	ddmmyy
2	ddyymm	18	ddyymm
3	yymmdd	19	yyddmm
4	yyddmm	20	mmddy
5	mmddy	21	mmyy
6	mmyy	22	ddmmyyyy
7	ddmmyyyy	23	ddyyyy
8	ddyyyy	24	yyyymmdd
9	yyyymmdd	25	yyyddmm
10	yyyddmm	26	mmddy
11	mmddy	27	mmyyyy
12	mmyyyy	28	dddy
13	dddy	29	yyyddd
14	yyyddd	30	yydd
15	yydd	31	ddy
16	ddy		

Complete the Special Handling for Sending panel (TP30) as follows:

TP30
Special Handling for Sending - BEG06

Transaction ID : SAMPLEPOSEND
Loop ID . . . :
Segment ID . . : BEG

Standard definition of data element
Element ID . . : 323 Elem qualifier :
Sequence num . . : 6 Description . . : Purchase Order Date
Type : AN Req des . . . : M
Minimum length : 6 Maximum length : 6

Additional definition of data element for this mapping
Date edit 9
Validation table name . . : +
Translation table name . . : +
User exit routine name . . : +
Concatenation position . . :
Concatenation length . . :

Concatenating Data for a Data Element (Send Only)

As a special handling example, suppose that an application field contains only part of the data that is required for a data element. The other part of the data must be provided either with a literal or by a different application field. For this example, a data element will be created using a literal value and an application field. Suppose that the application field ORGIDCODE is used as the source for data element 67 (identification code) in the N1 segment. Also suppose that the first two bytes of data element 67 are the same value for this particular application, and therefore, to save space, the ORGIDCODE in the application data does not contain these first two bytes.

To provide a value for data element 67, follow these steps:

1. Map data element 67:
 - a. In the **Application field name** field, type **ORGIDCODE**.
 - b. In the **Literal** field, type **&IFDATA 10**.
 - c. In the **Special handling** field, type **Y**.
 - d. Press Enter.

```

TP10          Map Data Element - N104

Transaction ID : SAMPLEPOSEND
Loop ID . . . : 000100      Occurrence number 1
Segment ID . . : N1

Standard definition of data element
Element ID . . : 67          Elem qualifier :
Sequence num . : 4           Description . . : Identification Code
Type . . . . . : AN          Req des . . . . : C
Minimum length : 2           Maximum length : 17

Definition of data element for this mapping transaction
Application field name . . : orgidcode +
Literal (enter on next line)
&IFDATA 10
Accumulator/action . . . . / _ _ / _ _ / _ _ / _ _
Special handling (Y/N) . . : y +

```

The Special Handling for Sending panel (TP30) panel is displayed.

2. To indicate what special handling is required:
 - a. In the **Concatenation position** field, type **1**.
 - b. In the **Concatenation length**, type **2**.
 - c. Press Enter.

Mapping Your Application Data to an EDI Standard Transaction Set

```
TP30                      Special Handling for Sending - N104

Transaction ID : SAMPLEPOSEND
Loop ID . . . : 000100      Occurrence number 1
Segment ID . . : N1

Standard definition of data element
Element ID . . : 67          Elem qualifier :
Sequence num . : 4           Description . : Identification Code
Type . . . . . : AN          Req des . . . : C
Minimum length : 2           Maximum length : 17

Additional definition of data element for this mapping
Date edit . . . . . : —
Validation table name . . : ———— +
Translation table name . . : ———— +
User exit routine name . . : ———— +
Concatenation position . . : 1 —
Concatenation length . . . : 2 —
```

3. Repeat the mapping of data element 67 by following these steps:
 - a. In the **Application field name** field, type **ORGIDCODE**.
 - b. In the **Special handling** field, type **Y**.
 - c. Press Enter.

```
TP10                      Map Data Element - N104

Transaction ID : SAMPLEPOSEND
Loop ID . . . : 000100      Occurrence number 1
Segment ID . . : N1

Standard definition of data element
Element ID . . : 67          Elem qualifier :
Sequence num . : 4           Description . : Identification Code
Type . . . . . : AN          Req des . . . : C
Minimum length : 2           Maximum length : 17

Definition of data element for this mapping transaction
Application field name . . : orgidcode ———— +
Literal (enter on next line)

Accumulator/action . . . : / — — / — — / — — / —
Special handling (Y/N) . . : y +
```

The Special Handling for Sending panel (TP30) panel is displayed.

4. To indicate what type of special handling is required:
 - a. In the **Concatenation position** field, type **3**.
 - b. In the **Concatenation length**, type **4**.
 - c. Press Enter.

```

TP30                      Special Handling for Sending - N104

Transaction ID : SAMPLEPOSEND
Loop ID . . . : 000100      Occurrence number 1
Segment ID . . : N1

Standard definition of data element
Element ID . . : 67          Elem qualifier :
Sequence num . : 4           Description . : Identification Code
Type . . . . . : AN          Req des . . . : C
Minimum length : 2           Maximum length : 17

Additional definition of data element for this mapping
Date edit . . . . . : —
Validation table name . . : ———— +
Translation table name . . : ———— +
User exit routine name . . : ———— +
Concatenation position . . : 3 —
Concatenation length . . . : 4 —
    
```

This mapping would result with a 10 in positions 1 and 2 of data element 67, and the value of ORGIDCODE in positions 3, 4, 5, and 6 of data element 67.

Mapping Tip: This mapping would always generate a 6-byte value for data element 67. If ORGIDCODE contained a value of 'A', then data element 67 would have a value of '10A '. Because concatenation was used, the trailing blanks were not removed. For numeric data, if the value is shorter than the specified length, the value is padded with leading zeroes.

There are other ways this mapping could have been done:

- Use repeat mapping without using concatenation position and length. With this method, you would:
 - a. In the first mapping of data element 67, specify application field ORGIDCODE and the literal **&IFDATA 10**.
 - b. Repeat the mapping of data element 67. Specify application field ORGIDCODE.

This will concatenate the value 10 with the value of ORGIDCODE and strip the trailing blanks for ORGIDCODE because concatenation was not specified on the Special Handling for Sending panel (TP30).

Concatenate position and length provide absolute control of the resulting field, but concatenation takes place automatically any time more than one application field is mapped to a standard data element with the repeat mapping capability. If concatenate position and length are not specified, then each application field mapping is treated independently with regard to removing unnecessary data, such as leading zeros and trailing blanks. The value is then concatenated with the current value in the standard data element. Thus if two application fields with a length of 3 and values of 'A ' and 'B ', the value in the data element would be 'AB'. If you wanted the result to be 'A B', you would have to use concatenate position and length.

- Use a named variable to provide the value for data element 67:
 - a. In the first mapping of the data element, specify application field ORGIDCODE and the literal **&IFDATA &SET val67 10**.
 - b. Repeat the mapping of the data element. Specify application field ORGIDCODE and the literal **&IFDATA &SAVE val67,3,***.

Mapping Your Application Data to an EDI Standard Transaction Set

- c. Repeat the mapping of the data element. Specify the literal **&USE val67**. Do not specify an application field.

This will also strip the trailing blanks because concatenation was not used. In addition, data will be moved to data element 67 only once, and standard compliance will be done only once.

- Use a user exit to automatically prefix the value of ORGIDCODE with the constant value.

Substringing Data for an Application Field (Receive Only)

Suppose that data element 67 (identification code) in the N1 segment is used as the source for the application field ORGIDCODE. Also suppose that the first two bytes of data element 67 are the same value for this particular application, and therefore, to save space, the ORGIDCODE in the application data does not want these first two bytes.

To provide a value for ORGIDCODE, follow these steps:

1. Map data element 67.
 - a. In the **Application field name** field, type **ORGIDCODE**.
 - b. In the **Special handling** field, type **Y**.
 - c. Press Enter.

```
TP10          Map Data Element - N104

Transaction ID : SAMPLEPOSEND
Loop ID . . . : 000100      Occurrence number 1
Segment ID . . : N1

Standard definition of data element
Element ID . . : 67          Elem qualifier :
Sequence num . : 4           Description . : Identification Code
Type . . . . . : AN         Req des . . . : C
Minimum length : 2          Maximum length : 17

Definition of data element for this mapping transaction
Application field name . . orgidcode +
Literal (enter on next line)

Accumulator/action . . . . / _ _ / _ _ / _ _ / _
Special handling (Y/N) . . y +
```

The Special Handling for Receiving panel (TP31) panel is displayed.

2. To indicate what type of special handling is required:
 - a. In the **Substring position** field, type **3**.
 - b. In the **Substring length**, type **4**.
 - c. Press Enter.

This mapping would ignore the first two bytes of data element 67, and move positions 3, 4, 5, and 6 from data element 67 into ORGIDCODE.


```

TP31                Special Handling for Receiving - N104

Transaction ID : SAMPLEPOSEND
Loop ID . . . : 000100      Occurrence number 1
Segment ID . . : N1

Standard definition of data element
Element ID . . : 67          Elem qualifier :
Sequence num . : 4           Description . : Identification Code
Type . . . . . : AN          Req des . . . : C
Minimum length : 2           Maximum length : 17

Additional definition of data element for this mapping
Date edit . . . . . : —
Validation table name . . : ———— +
Translation table name . . : ———— +
User exit routine name . . : ———— +
Substring position . . . : 3 —
Substring length . . . . : 4 —
Segment/loop qualifier . . : ————
    
```

Using Named Variables and Special Operators

Suppose that your application data contained information in a different format than specified by the EDI standard. You can use special operators to modify the format of the data. For example, if your application data stores phone numbers in the format (ABC)DEF-HIJK, and the EDI standard specifies phone numbers as ABCDEFHIJK, you can save the data in a named variable, then convert it using the TO special operator to remove the parentheses and hyphen for a send transaction or add the parentheses and hyphen for a receive transaction.

In both cases, you must map the data element twice. The first map saves the data to a named variable, and the second map converts the data to the required form.

Send Transaction: For the first mapping of the data element, you would complete the Map Data Element panel (TP10) as follows:

```

TP10                Map Data Element - PER04

Transaction ID : SAMPLEPOSEND
Loop ID . . . :
Segment ID . . : PER

Standard definition of data element
Element ID . . : 364          Elem qualifier :
Sequence num . : 4           Description . : Communication Number
Type . . . . . : DT          Req des . . . : 0
Minimum length : 6           Maximum length : 9

Definition of data element for this mapping transaction
Application field name . . : orgphone +
Literal (enter on next line)
&SAVE orgphone
Accumulator/action . . . : / — — / — — / — — / —
Special handling (Y/N) . . : N +
    
```

Mapping Your Application Data to an EDI Standard Transaction Set

For the second mapping of the data element, you would complete the Map Data Element panel (TP10) as follows:

TP10	Map Data Element - PER04
Transaction ID : SAMPLEPOSEND	
Loop ID . . . :	
Segment ID . . : PER	
Standard definition of data element	
Element ID . . :	364
Sequence num . :	4
Type :	DT
Minimum length :	6
Elem qualifier :	
Description . . :	Communication Number
Req des :	0
Maximum length :	9
Definition of data element for this mapping transaction	
Application field name . .	_____ +
Literal (enter on next line)	
&E(orgphone IS '(ABC)DEF-HIJK' TO 'ABCDEFHIJK')	
Accumulator/action	____ / ____ / ____ / ____ / ____ / ____
Special handling (Y/N) . .	N +

Mapping Tip: There are other ways to remove the parentheses and the dash from the application data. The method shown above is the easiest and clearest way to prepare data for the standard. The following options are also possible, and you may find them useful in other situations.

1. You could remove the characters that are not wanted by using the string division capability of DataInterchange. The expression **&E(orgphone / '(' / ')' / '-')** will also remove the unwanted characters. This method could prove useful if the unwanted characters are not in fixed positions within the application data.
2. You could specify a user exit, and the user exit could remove the unwanted characters and return the result to DataInterchange.
3. You could use a slight modification of the example shown by using the expression **&E(orgphone TO 'BCDFGHJKLM')**. This makes use of the fact that all variables have a default pattern of 'ABCDEFGHIJKLMNOPQRSTUVWXYZ'. This eliminates the IS operator, which reduces the length of the expression, allowing more to be specified on a single mapping. The disadvantage is that it is not as clear what data is being removed.

Receive Transaction: For the first mapping of the data element, you would complete the Map Data Element panel (TP10) as follows:

TP10	Map Data Element - PER04
Transaction ID : SAMPLEPOSEND	
Loop ID . . . :	
Segment ID . . : PER	
Standard definition of data element	
Element ID . . :	364 Elem qualifier :
Sequence num . :	4 Description . : Communication Number
Type :	DT Req des . . . : 0
Minimum length :	6 Maximum length : 9
Definition of data element for this mapping transaction	
Application field name . .	+ _____
Literal (enter on next line)	
&SAVE orgphone	
Accumulator/action	___/___/___/___/___/___
Special handling (Y/N) . .	N +

For the second mapping of the data element, you would complete the Map Data Element panel (TP10) as follows:

TP10	Map Data Element - PER04
Transaction ID : SAMPLEPOSEND	
Loop ID . . . :	
Segment ID . . : PER	
Standard definition of data element	
Element ID . . :	364 Elem qualifier :
Sequence num . :	4 Description . : Communication Number
Type :	DT Req des . . . : 0
Minimum length :	6 Maximum length : 9
Definition of data element for this mapping transaction	
Application field name . .	orgphone + _____
Literal (enter on next line)	
&FORCE &E(orgphone IS 'ABCDEFHIJK' TO '(ABC)DEF-HIJK')	
Accumulator/action	___/___/___/___/___/___
Special handling (Y/N) . .	N +

Specifying Occurrences

Within the EDI standards, certain segments and groups of segments can repeat. When you map a segment or loop that repeats within an EDI standard, you must tell DataInterchange which occurrence of the segment or loop you are using. This is known as qualifying the segment or loop.

When you qualify the loop, you need to know if the occurrence is a single-occurrence or a multiple-occurrence. Single-occurrence means that a specific instance of the segment or loop is mapped to a different area of the application data format. Multiple-occurrence means that all occurrences of the segment or loop are mapped to the same repeating application data format structure.


The Qualify Segment Mappings panel (TP07) or the Qualify Loop Mapping panel (TP08) is displayed for you to qualify the segments or loops when you map a segment or loop that repeats.

Qualifying a Segment

To see how to qualify a segment with occurrence numbers, look at the illustrated sample receive transaction. When you look at the incoming data, you see that multiple iterations of the N1 segment will be received. Through the trading partner agreements, the first occurrence of N1 will contain bill-to information and the second occurrence will contain ship-to information.

To map the N1 segment, follow these steps:

1. On the Transaction Segments panel (TP06), map the N1 segment.

Block	Copy	Delete mapping	Line	Map	Qualify	Repeat
TP06 Transaction Segments 1 to 8 of 66						
Transaction ID : SAMPLEPORECV						
Standard ID : X12V2400						
Standard transaction ID . . . : 850						
Mapped segments are highlighted.						
Ref	Segment	Seg/Loop	Req	Max	Loop	Loop
A	Num	ID	Des	Use	ID	Repeat
—	001	BEG	1	M	0001	0
—	002	NTE		F	0100	0
—	003	CUR		0	0001	0
—	004	REF		0	0012	0
—	005	PER		0	0003	0
—	006	TAX		0	0003	0
—	007	FOB		0	0001	0
	008	N1		0	0001	000100 200
						Description
						Beginning Segment (Purc
						Note/Special Informatio
						Currency
						Reference Numbers
						Administrative Communic
						Sales Tax Reference
						F.O.B. Related Informat
						Name

The Qualify Segment Mappings panel (TP07) is displayed. You need to indicate the first occurrence of the N1 segment is bill-to information.

Note: You can use the Prompt function to set the list of valid structure names, or look at the Application Data Format. For more information, refer to the section “Prompt (F4)” on page 2-11.

2. Type **1** in the **Occurrence number** field, and press Enter.

TP07 Qualify Segment Mapping	
For a single occurrence, enter the element ID (receive only) or occurrence number. For multiple occurrences, enter the structure name.	
Transaction ID : SAMPLEPORECV	
Standard transaction ID . . . : 850	
Loop ID :	
Segment ID : N1	
Segment description. : Name	
Element ID :	
Occurrence number	1
Structure name	+

The Data Elements panel (TP09) is displayed for you to map the bill-to information.

3. On the Transaction Segments panel (TP06), use **Repeat** to map the N1 segment again. The Qualify Segment Mappings panel (TP07) is displayed.

This time you need to indicate the second occurrence of the segments contains ship-to information.

4. Type **2** in the **Occurrence number** field, and press Enter.

The Data Elements panel (TP09) is displayed for you to map the ship-to information.

Qualifying a Loop

To see how you qualify a loop, look at the illustrated sample receive transaction. If you look at the incoming data, you will receive multiple iterations of the N1 segment. You have to tell DataInterchange how to recognize the start of a new N1 segment. Through the trading partner agreements, it has been established that data element 66 of the N1 segment will contain a **BT** to indicate the start of a BILLTO data format.

To map the N1 segment, follow these steps:

1. On Transaction Segments panel (TP06), map the N1 segment.

Block	Copy	Delete mapping	Line	Map	Qualify	Repeat		
TP06 Transaction Segments 8 to 15 of 66								
Transaction ID : SAMPLEPORECV								
Standard ID : X12V2400								
Standard transaction ID . . : 850								
Mapped segments are highlighted.								
A	Ref	Segment	Seg/Loop	Req	Max	Loop	Loop	Description
	Num	ID	Mappings	Des	Use	ID	Repeat	
m	008	N1		0	0001	000100	200	Name
—	009	N2	*****	0	0002	000100	0	Additional Name Informa
—	010	N3	*****	0	0002	000100	0	Address Information
—	011	N4	*****	0	0001	000100	0	Geographic Location
—	012	REF	*****	0	0012	000100	0	Reference Numbers
—	013	PER	*****	0	0003	000100	0	Administrative Communic
—	014	FOB	*****	0	0001	000100	0	F.O.B. Related Informat
—	015	CTP		0	0025	0	0	Pricing Information

The Qualify Loop Mapping panel (TP08) is displayed for you to qualify this occurrence of the loop.

Note: You can use the Prompt function to set the list of valid structure names, or look at the Application Data Format. For more information, refer to the section “Prompt (F4)” on page 2-11.

2. Because the trading partner agreement is specified using data element 66 of the N1 segment, type the data element ID, and press Enter.

TP08		Qualify Loop Mapping	1 to 7 of 7
For a single occurrence, enter the element ID (receive only) or occurrence number. For multiple occurrences, enter the structure name.			
Transaction ID : SAMPLEPORECV			
Standard transaction ID . . : 850			
Loop ID : 000100			
Element ID : 66			
Occurrence number : _____			
Structure name : _____ +			
Segments in Loop	Description		
N1	Name		
N2	Additional Name Information		
N3	Address Information		
N4	Geographic Location		
REF	Reference Numbers		
PER	Administrative Communications Contact		
FOB	F.O.B. Related Information		

Mapping Your Application Data to an EDI Standard Transaction Set

The Loop Segments panel (TP14) is displayed, listing the segments in this loop.

3. Map the appropriate segments. The action bar of TP14 changes to include the action “Float delete.” The actions are listed in alphabetical order.

Block	Copy	Delete mapping	Line	Map	Qualify	Repeat		
TP14			Loop Segments			1 to 7 of 7		
Transaction ID : SAMPLEPORECV								
Standard transaction ID . . : 850								
Loop ID : 000100 Qualifier = ** Not mapped **								
Mapped segments are highlighted.								
A	Ref Num	Segment ID	Seg/Loop Mappings	Req Des	Max Use	Inner Loop ID	Loop Repeat	Description
m	001	N1		0	0001	0	0	Name
	002	N2		0	0002	0	0	Additional Name Inform
m	003	N3		0	0002	0	0	Address Informati
m	004	N4		0	0001	0	0	Geographic Locati
-	005	REF		0	0012	0	0	Reference Numbers
-	006	PER		0	0003	0	0	Administrative Communi
	007	FOB		0	0001	0	0	F.O.B. Related Informa

The Data Elements panel (TP09) is displayed.

4. Map the appropriate elements; type **M** in the action column next to the elements you want to map, and press Enter.

Delete mapping	Line	Map	Qualify	Repeat	Select		
TP09			Data Elements		1 to 4 of 4		
Transaction ID . . . : SAMPLEPORECV							
Loop ID		000100	Qualifier = ** Not mapped **				
Segment ID		N1					
Mapped data elements are highlighted. ***nnn points to a qualifying element.							
A	Seq Num	Element ID	Element Mappings	Req Des	Rel Type Def	Related Element Ref Numbers	Description
-	1	98		M	ID		Organization Identifi
	2	93		C	AN		Name
m	3	66		C	ID		Identification C
	4	67		C	AN		Identification Code

Note: Normally, you would map element ID 98 first, but it was omitted here to show you how to finish qualifying the loop.

The Map Data Element panel (TP10) is displayed for data element 66.

5. You must enter a Y in the **Special handling** field, instead of the normal N, so that when you press Enter, the Special Handling for Receiving panel (TP31) will be displayed.

6. On the Special Handling for Receiving panel (TP31), you can specify the loop qualifier.

TP31
Special Handling for Receiving - N103

Transaction ID : SAMPLEPORECV
Loop ID . . . : 000100 Qualifier = ** Not mapped **
Segment ID . . : N1

Standard definition of data element
Element ID . . : 66 Elem qualifier :
Sequence num . : 3 Description . : Identification Code Qualifier
Type : ID Req des . . . : C
Minimum length : 1 Maximum length : 2

Additional definition of data element for this mapping
Date edit :
Validation table name . : +
Translation table name . : +
User exit routine name . : +
Substring position . . . :
Substring length :
Segment/loop qualifier . : **BT**

Note: If the qualifying data element is used with conditional processing and repeated mappings of this data element are needed, the conditional processing will only be executed if the first occurrence of the qualifying data element specifies a literal or application data field in the mapping.

Qualifying a Data Element

If you have data elements that repeat, you may need to qualify how DataInterchange handles each occurrence. For example, if your receive data has multiple units of measurements (each, case, and so forth), you may want to qualify information based on EA for each.

To qualify a data element, follow these steps:

1. On the Data Elements panel (TP09), qualify data element 355 and select data elements 330 and 212 as shown.

Delete mapping Line Map Qualify Repeat Select

TP09
Data Elements
1 to 8 of 25

Transaction ID . . . : SAMPLEPORECV
Loop ID : 000200 Multiple occurrence : LINEITEMS
Segment ID : P01

Mapped data elements are highlighted. ****nnn points to a qualifying element.

A	Seq Num	Element ID	Element Mappings	Req Des	Type	Rel Def	Related Element Ref Numbers	Description
	1	350		O	AN			Purchase Order Line N
s	2	330	1	M	R			Quantity Ordered
q	3	355	1	M	ID			Unit of Measurement C
s	4	212	1	C	R			Unit Price
	5	639		O	ID			Basis of Unit Price C
—	6	235	1	O	ID			Product/Service ID Qu
—	7	234	1	C	AN			Product/Service ID
—	8	235		O	ID			Product/Service ID Qu

The Data Element Qualifier panel (TP32) is displayed.

Mapping Your Application Data to an EDI Standard Transaction Set

2. In the **Qualifier** field, type **BF**, and press Enter.

Mapping Tip: Qualifiers are case sensitive: **BF** is not the same as **bf**.

TP09		Data Elements	1 to 8 of 25
Transaction ID . . . : SAMPLEPORECV			
Loop ID : 000200 Multiple occurrence : LINEITEMS			
Segment ID : P01			
Mapped data elements are highlighted. ***nnn points to a qualifying element.			
Seq		TP32	Data Element Qualifier
A	Num		
—	1	Enter the value to be used to qualify this element and	
s	2	associated elements. Press enter with no qualifier to	
q	3	end the qualifications.	
s	4		
—	5	Element ID : 355	N
—	6	Qualifier . . BF	C
—	7		
—	8		

The Data Element Qualifier panel (TP32) is redisplayed. :cp 10

3. In the **Qualifier** field, type **IN**, and press Enter.

TP09		Data Elements	1 to 8 of 25
Transaction ID . . . : SAMPLEPORECV			
Loop ID : 000200 Multiple occurrence : LINEITEMS			
Segment ID : P01			
Mapped data elements are highlighted. ***nnn points to a qualifying element.			
Seq		TP32	Data Element Qualifier
A	Num		
—	1	Enter the value to be used to qualify this element and	
s	2	associated elements. Press enter with no qualifier to	
q	3	end the qualifications.	
s	4		
—	5	Element ID : 355	N
—	6	Qualifier . . IN	C
—	7		
—	8		

The Data Element Qualifier panel (TP32) is redisplayed.

4. In the **Qualifier** field, type **RD**, and press Enter.

Mapping Your Application Data to an EDI Standard Transaction Set

TP09
Data Elements
1 to 8 of 25

Transaction ID . . . : SAMPLEPORECV
Loop ID : 000200 Multiple occurrence : LINEITEMS
Segment ID : P01

Mapped data elements are highlighted. ****nnn points to a qualifying element.

Seq
A Num

TP32
Data Element Qualifier

1
— 2
q 3
s 4
— 5
— 6
— 7
— 8

Enter the value to be used to qualify this element and associated elements. Press enter with no qualifier to end the qualifications.

Element ID : 355
Qualifier . . **RD**

N
C
C

The Data Element Qualifier panel (TP32) is redisplayed.

- Press Enter to indicate that you are done entering values.

The Data Elements panel (TP09) is displayed.

- Map data element 330; type **M** in the action column next to Element ID 330, and press Enter.

Delete mapping	Line	Map	Qualify	Repeat	Select	
TP09						Data Elements 1 to 8 of 25
Transaction ID . . .						SAMPLEPORECV
Loop ID						000200 Multiple occurrence : LINEITEMS
Segment ID						P01
Mapped data elements are highlighted. ****nnn points to a qualifying element.						
A	Seq Num	Element ID	Element Mappings	Req Des	Rel Type Def	Related Element Ref Numbers Description
	1	350		O	AN	Purchase Order Line N
m	2	330	*** 3	M	R	Quantity Ordered
—	3	355	*** 3	M	ID	Unit of Measurement C
—	4	212	*** 3	C	R	Unit Price
—	5	639		O	ID	Basis of Unit Price C
—	6	235	1	O	ID	Product/Service ID Qu
—	7	234	1	C	AN	Product/Service ID
—	8	235		O	ID	Product/Service ID Qu

The Qualified Data Element Mappings panel (TP15) is displayed, listing the qualifier values.

- Map the BF qualifier value; type **M** in the action column next to Element Qualifier BF, and press Enter.

Mapping Your Application Data to an EDI Standard Transaction Set

Add Map Repeat Delete mapping			
TP15		Qualified Data Elements Mappings - P0102	
		1 to 3 of 3	
Transaction ID : SAMPLEPORECV			
Segment ID : P01			
Data element ID : 330			
Type : R			
Minimum length : 1			
Maximum length : 9			
	Ref		
Action	Num	Element Qualifier	Mappings
m	1	BF	0
-	2	IN	0
-	3	RD	0

The Map Data Element panel (TP10) is displayed. The **Element qualifier** field on this panel shows the current value being mapped. In this case, the value is BF.

8. In the **Application field name** field, type **QTY**, and press Enter.

TP10		Map Data Element - P0102	
Transaction ID : SAMPLEPORECV			
Loop ID . . . :			
Segment ID . . : P01			
Standard definition of data element			
Element ID . . : 330		Elem qualifier : BF	
Sequence num . : 2		Description . : Quantity Ordered	
Type : R		Req des . . . : M	
Minimum length : 1		Maximum length : 9	
Definition of data element for this mapping transaction			
Application field name . . qty +			
Literal (enter on next line)			
Accumulator/action . . . / _ _ / _ _ / _ _ / _			
Special handling (Y/N) . . N +			

The Qualified Data Element Mappings panel (TP15) is redisplayed.

9. Map the IN qualifier value; type **M** in the action column next to Element Qualifier IN, and press Enter.

Add	Map	Repeat	Delete	mapping
TP15	Qualified Data Elements Mappings - P0102			1 to 3 of 3
Transaction ID : SAMPLEPORECV				
Segment ID : P01				
Data element ID : 330				
Type : R				
Minimum length : 1				
Maximum length : 9				
Action	Ref	Num	Element Qualifier	Mappings
		1	BF	1
m		2	IN	0
-		3	RD	0

The Map Data Element panel (TP10) is displayed.

10. In the **Literal** field, type **&SAVE QTY**, and press Enter.

TP10	Map Data Element - P0102	
Transaction ID : SAMPLEPORECV		
Loop ID . . . :		
Segment ID . . : P01		
Standard definition of data element		
Element ID . . : 330	Elem qualifier : IN	
Sequence num . : 2	Description . : Quantity Ordered	
Type : R	Req des . . . : M	
Minimum length : 1	Maximum length : 9	
Definition of data element for this mapping transaction		
Application field name +		
Literal (enter on next line)		
&SAVE QTY		
Accumulator/action / _ / _ / _ / _ / _		
Special handling (Y/N) . . . N +		

The Qualified Data Element Mappings panel (TP15) is redisplayed.

11. Use **Repeat** to map the IN qualifier value again.

Add	Map	Repeat	Delete	mapping
TP15	Qualified Data Elements Mappings - P0102			1 to 3 of 3
Transaction ID : SAMPLEPORECV				
Segment ID : P01				
Data element ID : 330				
Type : R				
Minimum length : 1				
Maximum length : 9				
Action	Ref	Num	Element Qualifier	Mappings
		1	BF	1
r		2	IN	1
-		3	RD	0

The Map Data Element panel (TP10) is displayed.

Mapping Your Application Data to an EDI Standard Transaction Set

12. Do the following:

- In the **Application field name** field, type **QTY**.
- In the **Literal** field, type **&FORCE &E(QTY/144)**. This forces a modified value into the application field.
- Press Enter.

```

TP10          Map Data Element - P0102

Transaction ID : SAMPLEPORECV
Loop ID      :
Segment ID   : P01

Standard definition of data element
Element ID   : 330      Elem qualifier : IN
Sequence num : 2        Description  : Quantity Ordered
Type        : R         Req des     : M
Minimum length : 1      Maximum length : 9

Definition of data element for this mapping transaction
Application field name . . QTY +
Literal (enter on next line)
&FORCE &E(QTY/144)
Accumulator/action . . . / _ / _ / _ / _ / _
Special handling (Y/N) . . N +
  
```

The Qualified Data Element Mappings panel (TP15) is redisplayed.

13. Map the RD qualifier value; type **M** in the Action column next to Element Qualifier RD, and press Enter.

```

Add Map Repeat Delete mapping

TP15          Qualified Data Elements Mappings - P0102          1 to 3 of 3

Transaction ID . . . . . : SAMPLEPORECV
Segment ID . . . . . : P01
Data element ID . . . . . : 330
Type . . . . . : R
Minimum length . . . . . : 1
Maximum length . . . . . : 9

Action      Ref      Element Qualifier      Mappings
  -          1      BF                      1
             2      IN                      2
  M          3      RD                      0
  
```

The Map Data Element panel (TP10) is displayed.

14. In the **Literal** field, type **&SAVE QTY**, and press Enter.

```

TP10          Map Data Element - P0102

Transaction ID : SAMPLEPORECV
Loop ID      :
Segment ID   : P01

Standard definition of data element
Element ID   : 330      Elem qualifier : RD
Sequence num : 2        Description   : Quantity Ordered
Type        : R         Req des      : M
Minimum length : 1      Maximum length : 9

Definition of data element for this mapping transaction
Application field name . . . . . +
Literal (enter on next line)
&SAVE QTY
Accumulator/action . . . . . _ / _ _ / _ _ / _ _ / _
Special handling (Y/N) . . . . . N +
    
```

The Qualified Data Element Mappings panel (TP15) is redisplayed.

15. Use **Repeat** to map the RD qualifier value again.

```

Add  Map  Repeat  Delete mapping
-----
TP15          Qualified Data Elements Mappings - P0102          1 to 3 of 3

Transaction ID . . . . . : SAMPLEPORECV
Segment ID . . . . . : P01
Data element ID . . . . . : 330
Type . . . . . : R
Minimum length . . . . . : 1
Maximum length . . . . . : 9

Action      Ref      Element Qualifier      Mappings
-----
-           1      BF           1
r         2      IN           2
           3      RD           1
    
```

The Map Data Element panel (TP10) is displayed.

16. Do the following:

- In the **Application field name** field, type **QTY**.
- In the **Literal** field, type **&FORCE &E(QTY*198/144)**. This forces a modified value into the application field.
- Press Enter.

Mapping Your Application Data to an EDI Standard Transaction Set

```
TP10          Map Data Element - P0102

Transaction ID : SAMPLEPORECV
Loop ID      . . . :
Segment ID   . . . : P01

Standard definition of data element
Element ID . . . : 330      Elem qualifier : IN
Sequence num . . . : 2      Description . . : Quantity Ordered
Type . . . . . : R          Req des . . . . : M
Minimum length : 1          Maximum length : 9

Definition of data element for this mapping transaction
Application field name . . QTY +
Literal (enter on next line)
&FORCE &E(QTY*198/144)
Accumulator/action . . . _ / _ _ / _ _ / _ _ / _
Special handling (Y/N) . . N +
```

Specifying Trading Partners

Once you have completed mapping your send and receive transactions, you have to inform DataInterchange which transactions correspond to which trading partner. You can specify the same transaction mappings for several trading partners and provide specific overrides for each trading partner. This enables you to modify a transaction mapping from one trading partner to the next.

| **Note:** If you are using the Minimal Trading Partners feature described in chapter 5, then the specification
| of trading partners may not be required. See “Overview of the Minimal Trading Partners Feature” on
| page 5-2, for more information.

Add Trading Partner Usage for Sending

The Add Trading Partner Usage for Sending panel (TP17) is displayed: when you use the **Where used** action on the Transaction Mappings panel (TP01), when you are mapping a send transaction, and when no trading partner usages are defined for this mapping. From the list of trading partner usages you can:

- Activate a trading partner usage
- Add a new trading partner usage
- Copy a trading partner usage and change it
- Delete a trading partner usage
- Migrate a trading partner usage
- Print a trading partner usage
- Update a trading partner usage
- Display a list of transaction mappings associated with a trading partner usage (**Where used**)

TP17

Add Trading Partner Usage for Sending

Transaction ID : SAMPLEPOSEND
Data format ID : SAMPLEPOSEND

Sending TP nickname
Receiving TP nickname
Internal trading partner ID
Post-translation exit routine
Group encryption key name
Group authentication key name
Trans encryption key name
Trans authentication key name
Acknowledgment expected (Y/N)
Activate usage (Y/N)
Usage indicator (P/T/I)
Log application data (Y/N)

+
+

+

- +
- +
- +
- +

The following list describes the fields of the Add Trading Partner Usage for Sending panel.

In this field:

Transaction ID

Data format ID

Sending TP
nickname

Enter:

The ID of this transaction mapping. The system supplies this value for your reference.

The data format you are using for this transaction mapping. The system supplies this value for your reference.

The name of the sending trading partner as you entered it in the Trading Partner Profile, or either of the special keywords ANY or KNOWN. The sending trading partner's interchange ID and qualifier will appear in the envelope header segment (ISA for X12 and UNB for EDIFACT) as the interchange sender ID and interchange sender qualifier.

You may use the special keywords ANY and KNOWN for the sending trading partner name. The value ANY means that this usage can be used by any sending trading partner, even if that trading partner is not known to DataInterchange. The value KNOWN means that this usage can be used by any trading partner defined in the Trading Partner Profile.

For more information on the ANY and KNOWN keywords, see "Overview of the Minimal Trading Partners Feature" on page 5-2.

Mapping Your Application Data to an EDI Standard Transaction Set

In this field:	Enter:
Receiving TP nickname	<p>The name of the receiving trading partner as you entered it in the Trading Partner Profile, or either of the special keywords ANY or KNOWN. The receive trading partner's interchange ID and qualifier will appear in the envelope header segment (ISA for X12 and UNB for EDIFACT) as the interchange receiver ID and interchange receiver qualifier.</p> <p>You may use the special keywords ANY and KNOWN for the receiving trading partner name. The value ANY means that receiving trading partner name. The value ANY means that this usage can be used by any receiving trading partner, even if that trading partner is not known to DataInterchange. The value KNOWN means that this usage can be used by any trading partner defined in the Trading Partner Profile.</p> <p>When defining a generic send usage, leave the trading partner nickname field blank.</p> <p>To define a generic receive usage, enter an ampersand (&) only in the Trading Partner Nickname. The generic receive usage is selected when a specific active usage by trading partner nickname is not found and the Allow Generic Recv field in the trading partner profile is set to "Y."</p> <p>For more information on the ANY and KNOWN keywords, see "Overview of the Minimal Trading Partners Feature" on page 5-2.</p>
Internal trading partner ID	<p>The trading partner who receives this transaction from you. This is usually a vendor or customer number that your application uses to refer to the trading partner. You can use any combination of characters in the ALPHANUM table.</p> <p>You can define a generic send usage to be used for multiple trading partners by entering an ampersand (&) in the internal trading partner ID field followed by the 3-character generic routing code provided by the application, or blanks for a default generic usage. The specific generic usage by routing code or the default usage is selected when no specific usage by Internal trading partner ID is found. When generic usages are defined, a translate table with the same name as the Data format ID must also be defined to contain the Internal trading partner ID to trading partner nickname relationship. See "Defining Generic Send Usages" on page 9-85 for more information.</p> <p>The Internal trading partner ID and Data format ID form the primary key the translator uses to find the usage that determines appropriate mapping. If you have more than one send usage defined for this trading partner and data format (for example, test and production usages), you must associate each usage with a different Transaction ID.</p> <p>When using C and D records, this value is specified in the control (C) record. When using raw data, this value is in the field specified in the Trading partner ID field in the application data format. See the description of this field on page 7-10 for more information.</p>
Post-translation exit routine	<p>The user-written exit routine that the translator calls after translating the transaction. The value must match the name of a member of the ADAMCTL profile. This field is optional.</p>
Group encryption key name	<p>The encryption key that the translator puts in the group security segments and passes to a user-written encryption routine. Any combination of A—Z and 0—9 is valid. This field is optional.</p>

In this field:

Group authentication key name

Trans encryption key name

Trans authentication key name

Acknowledgment expected

Activate usage

Enter:

The authentication key that the translator puts in the group security segments and passes to a user-written authentication routine. You can use any combination of characters in the ALPHANUM table. This field is optional.

The encryption key that the translator puts in the transaction set security segments and passes to a user-written encryption routine. You can use any combination of characters in the ALPHANUM table. This field is optional.

The name of the authentication key that the translator puts in the transaction set security segments and passes to a user-written authentication routine. You can use any combination of characters in the ALPHANUM table. This field is optional.

Note: For more information about exit routines, see the *DataInterchange Programmer's Reference*. For related profile information, see "User Program Information Profile (ADAMCTL)" on page 4-5 and "Security Profile (SECUPROF)" on page 4-9

The code that indicates whether you expect to receive functional acknowledgments for this transaction. Valid values are Y (yes) and N (no).

The code that indicates whether the usage you are defining is the active usage for this internal trading partner and data format combination. Valid values are Y (yes) and N (no).

This field works with the *Test usage* field to provide one active usage for production and one active usage for testing. Because only one of each (for an internal trading partner and data format combination) can be active at a time, entering Y deactivates the current usage if one of this type is active. The valid combinations are:

Active	Test	Usage
Y	Y	Choice 1 for testing
Y	N	Production and choice 2 for testing
N	Y	Not used
N	N	Not used

Usage indicator

Enter P (for production), T (for test), or I (for information) to indicate the type of usage.

The usual mode of operation for a production message is that a production usage will be used if an active production usage exists. Otherwise, an error will occur.

The usual mode of operation for a test message is that a test usage will be used if an active test usage exists. Otherwise, a production usage will be used if an active production usage exists. Otherwise, an error will occur.

The usual mode of operation for an information message is that an information usage will be used if an active information usage exists. Otherwise, a production usage will be used if an active production usage exists. Otherwise, an error will occur.

Log application data

The code that indicates whether you want the translator to save an image of this transaction in application format in the event log. Valid values are Y (yes) or N (no).

Trading Partner Usage Overrides for Sending

The Trading Partner Usage Overrides for Sending panel (TP26) is displayed after you enter the usage information on the Add Trading Partner Usage for Sending panel (TP17). Use this panel to further define the trading partner usage.

Note: All the fields on this panel are optional, but if you enter a value here, it overrides the value that the translator normally uses.

TP26
Trading Partner Usage Overrides for Sending

Transaction ID : SAMPLEPOSEND
Data format ID : SAMPLEPOSEND
Internal trading partner ID . . . : SAMPLEPARTNER

Envelope type (E/I/T/U/X/0) _ +
Standard envelope profile member . . . _____
Application sender ID _____
Application receiver ID _____
Application password _____
Group security profile member name . . _____
Trans security profile member name . . _____
Validation level (0/1/2) -
Acceptable error level (0/1/2) -
Enforce structure hierarchy (Y/N) . . . N +
Structure must produce data (Y/N) . . . N +
Control numbers by trans ID (Y/N) . . . N +
Alphanumeric data validation table . . _____
Character data validation table . . . _____

The following list describes the fields of the Trading Partner Usage Overrides for Sending panel (TP26).

In this field:	Enter:														
Transaction ID	The ID of this transaction mapping. The system supplies this value for your reference.														
Data format ID	The data format you are using for this transaction mapping. The system supplies this value for your reference.														
Internal trading partner ID	The ID of the trading partner who receives this transaction from you. The system supplies this value for your reference.														
Envelope type	<p>The type of enveloping to use when sending this transaction. Valid values are:</p> <table> <tr> <th>Value</th><th>Envelope Type</th></tr> <tr> <td>E</td><td>EDIFACT (UNB/UNZ)</td></tr> <tr> <td>I</td><td>Interchange Control Segments (ICS/ICE)</td></tr> <tr> <td>T</td><td>Trade Data Interchange (STX/END)</td></tr> <tr> <td>U</td><td>Uniform Communication Standard (BG/EG)</td></tr> <tr> <td>X</td><td>X12 (ISA/IEA)</td></tr> <tr> <td>0</td><td>None (no interchange envelope will be used)</td></tr> </table> <p>EDI standard I, U, and X transactions can be sent without an interchange envelope.</p> <p>A value entered here overrides the value in the standard.</p>	Value	Envelope Type	E	EDIFACT (UNB/UNZ)	I	Interchange Control Segments (ICS/ICE)	T	Trade Data Interchange (STX/END)	U	Uniform Communication Standard (BG/EG)	X	X12 (ISA/IEA)	0	None (no interchange envelope will be used)
Value	Envelope Type														
E	EDIFACT (UNB/UNZ)														
I	Interchange Control Segments (ICS/ICE)														
T	Trade Data Interchange (STX/END)														
U	Uniform Communication Standard (BG/EG)														
X	X12 (ISA/IEA)														
0	None (no interchange envelope will be used)														

In this field:

Standard profile member

Enter:

The envelope profile member to use when building envelopes for this transaction. The value must match the name of an existing envelope profile member.

You can define a generic standard profile member name by entering an ampersand (&) followed by a 1- to 6-character base name. At execution time, the envelope profile suffix from the trading partner profile is appended to the base name to dynamically generate the name of the standard profile member to use when enveloping the transactions. See “Defining Generic Send Usages” on page 9-85 and “Defining Generic Receive Usages” on page 9-96 for more information.

A value entered here overrides the envelope profile member with the same name as the standard ID used in the envelope.

Application sender ID

The specific sender within your company, such as a department number. You can define several different usages with different data formats to route transactions based on sender ID. The sender ID maps to the standard data element with data type AS.

A value entered here overrides the sender ID in the envelope profile member. Providing a sender ID in the control record passed to the translator overrides the sender ID in the envelope profile member and any value specified on this panel. This field is not used with 0 enveloping.

Application receiver ID

The specific receiver within the trading partner, such as a department number. You can define several different usages with different data formats to route transactions based on receiver ID. The receiver ID maps to the standard data element with data type AR.

A value entered here overrides the receiver ID in the envelope profile member. Providing a receiver ID in the control record passed to the translator overrides the receiver ID in the envelope profile member and any value specified on this panel. This field is not used with 0 enveloping.

Application password

The value used as the password in the functional group envelope. The password maps to standard data element with data type PW.

A value entered here overrides the password in the envelope profile member. Providing a password in the control record passed to the translator overrides the password in the envelope profile member and any value specified on this panel.

Group security profile member name

The security profile member that contains the information the translator needs at the functional group level to:

- Build security segments
- Call user-written exit routines for security and data compression

A value entered here overrides the security profile member name in the trading partner profile.

Trans security profile member name

The security profile member that contains the information the translator needs at the transaction set level to:

- Build security segments
- Call user-written exit routines for security and data compression

A value entered here overrides the security profile member name in the trading partner profile.

Mapping Your Application Data to an EDI Standard Transaction Set

In this field: Validation level	Enter: The code that indicates the level of validation done during translation. Valid values are: <table><tr><th>Value</th><th>Requests</th></tr><tr><td>0</td><td>No additional validation over the mandatory validation done by the translator to convert data between the application and standard data types. With this level of validation, any validation tables specified during mapping are ignored.</td></tr><tr><td>1</td><td>Level 0 validation plus the use of the validation tables specified during mapping.</td></tr><tr><td>2</td><td>Level 1 validation plus verification that the data values being supplied are consistent with the field's data type (for example, to verify that a DT field has a valid date, or a numeric field contains numeric data only).</td></tr></table> The default is 0. For more information about validation, see "Validation During Mapping" on page 9-37.	Value	Requests	0	No additional validation over the mandatory validation done by the translator to convert data between the application and standard data types. With this level of validation, any validation tables specified during mapping are ignored.	1	Level 0 validation plus the use of the validation tables specified during mapping.	2	Level 1 validation plus verification that the data values being supplied are consistent with the field's data type (for example, to verify that a DT field has a valid date, or a numeric field contains numeric data only).
Value	Requests								
0	No additional validation over the mandatory validation done by the translator to convert data between the application and standard data types. With this level of validation, any validation tables specified during mapping are ignored.								
1	Level 0 validation plus the use of the validation tables specified during mapping.								
2	Level 1 validation plus verification that the data values being supplied are consistent with the field's data type (for example, to verify that a DT field has a valid date, or a numeric field contains numeric data only).								
Acceptable error level	The code that indicates the acceptable level of errors for this trading partner transaction. Valid values are: <table><tr><th>Value</th><th>Sends</th></tr><tr><td>0</td><td>Transactions with no errors (default)</td></tr><tr><td>1</td><td>Transactions that have data element errors</td></tr><tr><td>2</td><td>Transactions that have segment errors</td></tr></table> The default is 0.	Value	Sends	0	Transactions with no errors (default)	1	Transactions that have data element errors	2	Transactions that have segment errors
Value	Sends								
0	Transactions with no errors (default)								
1	Transactions that have data element errors								
2	Transactions that have segment errors								
Enforce structure hierarchy	Y if you want the translator to issue an error if application structures are passed to DataInterchange out of the hierarchical sequence defined by the application data format, or N if you do not. The default is N. In either case DataInterchange will automatically create as many parent structures as necessary to satisfy the hierarchical definition, and initialize them with blanks. The translator will not issue an error for structures that only exist for grouping other structures. If a structure does not contain any fields, then its absence will not result in an error.								
Structure must produce data	Y if you want the translator to issue an error if either of the following occurs: <ul style="list-style-type: none">• An application structure associated with a loop or repeating segment was provided, but it did not generate any standard data as output; for example, if all the application fields in the structure mapped to the standard segment contained blank values. This only applies to structures that are passed separately as defined in the application data format.• An application structure is provided and that application structure is the sole source of data for a standard segment but no standard data was generated. Type N if you do not want an error issued in these situations. The default is N. Note: This field is not used if you are using hierarchical mapping support.								

In this field:	Enter:
Control numbers by trans ID	Y (for yes) or N (for no) to indicate whether or not you want the translator to use the standard transaction ID when updating control numbers. The trading partner profile allows you to add multiple pairs of control number information based on receiver ID, receiver qualifier, and standard transaction ID.
	Value
	Y Control numbers will be updated using the trading partner nickname, receiver ID, receiver qualifier, and standard transaction ID. If this trading partner combination is not found, the translation process will automatically create control number information for this combination using the trading partner profile member control number definition for the mask.
	N Control numbers will be updated using the trading partner nickname, receiver ID, and receiver qualifier.
Alphanumeric data validation tables	The name of the validation table you would like to use instead of ALPHANUM for alphanumeric validation. If supplied, this table will be used for all alphanumeric validation for transactions processed with this usage override. This field is optional.
Character data validation table	The name of the character set table you would like to use instead of CHARSET for character validation. If supplied, this table will be used for all character validation for transactions processed with this usage override. This field is optional.

Defining Generic Send Usages

Where multiple trading partners can use the same send usage definition and map, a generic send usage can be defined to DataInterchange. When combined with a generic routing code supplied by the application, it provides the capability to define one or more generic usages, each of which can handle multiple trading partners.

The generic routing code is an optional, 3-character code provided by the application to select the correct generic usage when no specific usage has been defined for the trading partner. The Generic Routing Code can be provided in one of three ways:

1. The Translator Control Block (TRCB) includes a 3-character field for the generic routing code. The API calls can provide the routing code in the TRCB.
2. For C and D processing, the C record includes a field for the 3-character generic routing code.
3. For raw data processing, an application field that contains the generic routing code can be specified when defining the data format. The application field should be from 1- to 3-characters. If less than three characters, the value is left adjusted with blank fill. If greater than three characters, it is truncated to three.

Defining Send Usages

When defining send usages, an internal trading partner ID that begins with an ampersand (&) indicates it is a generic usage. For a specific usage by routing code, specify an ampersand (&) followed by the 3-character generic routing code that the application will provide to select this usage.

If the application does not want to provide a routing code or does provide a routing code but wants to select a default generic usage, the internal TP ID can be defined as a single '&' followed by blanks. This

Mapping Your Application Data to an EDI Standard Transaction Set

type of generic definition is the default definition and is selected when no routing code is provided, or when the routing code is provided but no specific usage is found, or when the routing code is blank.

The purpose of the generic routing code is to allow a file containing one ADF to generate more than one type of transaction (such as purchase orders and purchase-order changes). The generic routing code is required when the user wants to process transactions using different maps that reference the same data format ID. For example, if the user wants to process purchase order and purchase-order change transactions that use the same data format, the application could provide the following: a routing code of POR for a purchase order and DataInterchange would select the usage with the internal trading partner ID of &POR, which would reference a purchase order map; and a routing code of POR for a purchase-order change and DataInterchange would select usage &POC, which would reference a purchase-order change map. An additional benefit when using this type of definition is that multiple transactions can be included in the same raw data file if they all use the same data format ID.

Determining the Trading Partner

Because the generic send usage can be used for many trading partners, the trading partner nickname in the send usage is ignored, and an alternate method is used to determine the trading partner nickname at run time.

During the usage selection process, if a generic send usage is selected, the applications internal trading partner ID must be related to a trading partner nickname. This is accomplished with a required user defined translate table as follows:

1. The translate table must be created by the user and named the same as the data format ID. This restricts the data format ID to 8 characters or less and if greater than 8 characters, it results in an error when defining generic send usages.
2. The translate table contains an internal trading partner ID (35-character local value) to trading partner nickname (16-character standard or trading partner value) relationship.
3. DataInterchange uses the internal trading partner ID to perform a search in this table to determine the trading partner nickname.

Selecting the Send Usage

DataInterchange locates the send usage through the following steps:

1. It first tries to find a send usage using the internal trading partner ID and data format ID and if found, it is the first choice for translation.

When translation is executed, a test indicator setting will affect the usage retrieved. The test indicator is set by the RAWTEST keyword on the PERFORM command, or it is contained in the Control Record. The test indicator has the following values and affects:

- Y - A test usage will be selected. If not found, a production usage will be selected and used as a test usage. If test or production usage is not found, an error will be reported.
 - N - A production usage will be selected. If not found, an error will be reported.
 - U - A test usage will be selected. If not found, a production usage will be selected. Both are used as selected.
2. If not found and the data format ID is 8 characters or less, the data format is used as the translate table name to look up the internal TP ID and relate it to a trading partner nickname. If the data format ID is greater than 8 characters, or the translate table is not defined, or the internal trading partner ID is not found in the table, a send usage not found error occurs.

3. If the search was successful, DataInterchange then retrieves the trading partner profile member using the trading partner nickname from the table and retrieves the send usages using the translation test indicator setting.

If the generic routing code is provided by the application and the generic routing code field is defined in the application data format, up to six usages could be retrieved.

- Production usage with internal trading partner ID of '&rrr' as a production usage with specific routing code
- Production usage with internal trading partner ID of '&rrr' as a test usage with specific routing code
- Test usage with internal trading partner ID of '&rrr' as a test usage with specific routing code
- Production usage with internal trading partner ID of '&' as the default production usage
- Production usage with internal trading partner ID of '&' as the default test usage with specific routing code
- Test usage with internal trading partner ID of '&' as the default test usage with specific routing code

If a generic routing code is **NOT** provided by the application or the generic routing code field is **NOT** defined in the application data format, up to three usages could be retrieved:

- Production usage with internal trading partner ID of '&' as the default production usage
- Production usage with internal trading partner ID of '&' as a test usage with specific routing code
- Test usage with internal trading partner ID of '&' as a test usage with specific routing code

Standard Envelope Profile Member Selection

When the send usage is found, the envelope profile member must be determined for enveloping information. The envelope profile member name can be specified in the send usage as a fixed name and must exist when the usage is defined. In this case the member name is used 'as is' to retrieve the envelope member.

Additionally, the envelope profile member name can be specified as a generic name, and the actual name is resolved at run time. To enable this feature, the *Envelope Profile Member Name* field on the send usage override panel allows the entry of a common (generic) member name (up to 6 characters) prefixed by an '&' (example: &ENPRID). The '&' is an indication to DataInterchange that the envelope profile member is to be selected at run time by appending the 2-character suffix (example: XX) from the Trading Partner Profile to the generic envelope profile member name (example: ENPRIDXX) and a call made to retrieve the member.

For generic envelope profile members, the Trading Partner Profile (TPPROF) definition includes a 2-character Generic Envelope Profile Suffix.

The use of the TPPROF suffix to select the envelope profile member allows the flexibility to have multiple envelope members (ENPRIDXX) and only one generic usage ('& '), or single envelope member (ENPRID) for multiple generic usages ('&rrr'), or a combination of these. This is most useful with, but not restricted to, generic usages.

These generic envelope profile members must be defined prior to performing enveloping or deenveloping.

Transaction Store Considerations

For tracking purposes, the actual trading partner nickname and internal TP ID are saved in the transaction store.

Management Reporting Considerations

For management reporting, statistics for all trading partners using a generic usage are collected under the usage internal trading partner ID. No statistics are collected by the actual trading partner or internal trading partner. If statistics are required by the trading partner, and generic send usages are being used, the transaction store is a better source for the statistics; however, it will require programming to accumulate the statistics.

Export/Import Considerations

When exporting generic send usages, the translate table containing the internal trading partner ID to trading partner nickname relationships is automatically exported, but the trading partner profile members referenced by the nicknames in the table are not exported and must be exported individually.

If a generic envelope profile member name is specified in the send usage override, the actual envelope profile members are unknown to export, and the members must be exported individually.

Add Trading Partner Usage for Receiving

The Add Trading Partner Usage for Receiving panel (TP22) is displayed: when you use the **Where used** action on the Transaction Mappings panel (TP01), when you are mapping a receive transaction, and when no trading partner usages are defined for this mapping.

TP22	Add Trading Partner Usage for Receiving
Transaction ID : SAMPLEPORECV	Standard transaction ID : 850
Sending TP nickname	_____ +
Receiving TP nickname	_____ +
Application routing (sender)	_____
Application routing (receiver)	_____
Responsible agency code	_____
Version	_____
Release	_____
Internal trading partner ID	_____
Pre-translation exit routine	_____
Acknowledgment type	_____ +
Group level FA only (Y/N)	_____ +
Inbound envelope used on FA (Y/N)	_____ +
Switch appl routing IDs on FA (Y/N)	_____ +
Activate usage (Y/N)	_____ +
Usage indicator (P/T/I)	_____ +
Log application data (Y/N)	_____ +

Mapping Your Application Data to an EDI Standard Transaction Set

The following list describes the fields of the Add Trading Partner Usage for Receiving panel.

In this field:

Enter:

Transaction ID

The ID of this transaction mapping. The system supplies this value for your reference.

Standard Transaction ID

The standard transaction set used for this transaction mapping. The system supplies this value for your reference.

Sending TP
nickname

Enter the name of the sending trading partner as you entered it in the Trading Partner Profile, or either of the special keywords ANY or KNOWN. The sending trading partner's interchange ID and qualifier will appear in the envelope header segment (ISA for X12 and UNB for EDIFACT) as the interchange sender ID and interchange sender qualifier.

You may use the special keywords ANY and KNOWN for the sending trading partner name. The value ANY means that this usage can be used by any sending trading partner, even if that trading partner is not known to DataInterchange. The value KNOWN means that this usage can be used by any trading partner defined in the Trading Partner Profile.

When defining a generic send usage, leave the trading partner nickname field blank.

To define a generic receive usage, enter an ampersand (&) only in the Sending TP Nickname. The generic receive usage is selected when a specific active usage by trading partner nickname is not found and the Allow Generic Recv field in the trading partner profile is set to 'Y'.

For more information on the ANY and KNOWN keywords, see "Overview of the Minimal Trading Partners Feature" on page 5-2.

Receiving TP
nickname

Enter the name of the receiving trading partner as you entered it in the Trading Partner Profile, or either of the special keywords ANY or KNOWN. The receive trading partner's interchange ID and qualifier will appear in the envelope header segment (ISA for X12 and UNB for EDIFACT) as the interchange receiver ID and interchange receiver qualifier.

You may use special keywords ANY and KNOWN for the receiving trading partner nickname. The value ANY means that this usage can be used by any receiving trading partner, even if that trading partner is not known to DataInterchange. The value KNOWN means that this usage can be used by any trading partner defined in the Trading Partner Profile.

For more information on the ANY and KNOWN keywords, see "Overview of the Minimal Trading Partners Feature" on page 5-2.

Mapping Your Application Data to an EDI Standard Transaction Set

In this field:

Application routing
(sender)

Enter:

The specific sender within the trading partner to route transactions based on sender ID (for example, a department number).

If you enter a value in this field, do not use the **Application routing (receiver)** field. When a transaction is received, the translator looks for a sender ID to determine the correct trading partner usage. If there is no sender ID, the translator looks for the receiver ID to determine the correct usage. If there is no receiver ID, the translator uses a blank ID to determine the correct usage. Therefore, you can define several active usages to route transactions based on either the sender or receiver ID, but not both.

- If functional groups are present in the interchange being received, the application sender ID is taken from the field with data type AS.
- If functional groups are not present, the application sender ID is taken from the field with data type RS.
- If the field with data type RS is defined but contains blanks or no data, the application sender ID is taken from the field with data type IS.
- If the field with data type RS is not defined, the application sender ID is taken from the field with data type AS.
- If the field with data type AS is not defined, the application sender ID is taken from the field with data type IS.

Application routing
(receiver)

The specific receiver within your company to route transactions based on receiver ID (for example, a department number).

If you enter a value in this field, do not use the **Application routing (sender)** field.

- If functional groups are present in the interchange being received, the application receiver ID is taken from the field with data type AR.
- If functional groups are not present, the application receiver ID is taken from the field with data type RR.
- If the field with data type RR is defined but contains blanks or no data, the application receiver ID is taken from the field with data type IR.
- If the field with data type RR is not defined, the application receiver ID is taken from the field with data type AR.
- If the field with data type AR is not defined, the application receiver ID is taken from the field with data type IR.

Responsible agency
code

Optional. This field may be used in addition to the application sender and application receiver IDs to provide additional routing capabilities. It is expected, but not necessary, that the agency code would be combined with the version and release (next two fields) so that different routing (or a completely different map) would be possible based on the defining agency and the version of the standard transaction. For example, it is possible to have a separate map for a TDCC 810 (agency code value of 'T' and an X12 810 (agency code of 'X') from the same trading partner. The data entered into this field must exactly match the agency code when taken from the received transaction other than trailing blanks, which are not significant.

In this field:

Version

Enter:

Optional. This field may be used in addition to the application sender and application receiver IDs to provide additional routing capabilities. It is expected, but not necessary, that version would be combined with the agency (previous) and release (next) so that different routing (or a completely different map) would be possible based on the defining agency and the version of the standard transaction. For example, it is possible to have a separate map for a TDCC Version 3 810 and an X12 Version 3 810 from the same trading partner. The data entered into this field must exactly match the version when taken from the received transaction other than trailing blanks, which are not significant.

Release

Optional. This field may be used in addition to the application sender and application receiver IDs to provide additional routing capabilities. It is expected, but not necessary, that release would be combined with the agency and version (previous two fields) so that different routing (or a completely different map) would be possible based on the defining agency, version and release of a transaction. For example, it is possible to have a separate map for an X1 Version 3 Release 1 810 transaction and an X12 Version 3 Release 2 810 transaction from the same trading partner. The data entered into this field must exactly match the release when taken from the received transaction other than trailing blanks, which are not significant.

Internal trading
partner ID

Optional. The trading partner who sends you this transaction. This is usually a vendor or customer number your application uses to refer to the trading partner. You can use any combination of characters in the ALPHANUM table.

This literal value is placed in the field specified, if any, in the **Trading partner ID field** of the application data format. See the description of this field on page 7-10 for more information.

For generic receive usages, this is the literal value placed in the application data for all transactions that select this usage for translation. For more information, see “Defining Generic Send Usages” on page 9-85 and “Defining Generic Receive Usages” on page 9-96.

Pre-translation exit
routine

The name of the user-written exit routine the translator calls before translating the transaction. The name must match the name of a member of the ADAMCTL profile. For information about exit routines, see the *DataInterchange Programmer's Reference*.

Acknowledgment type

The acknowledgment type determines the transaction ID and the version and release to be generated by DataInterchange. This field is optional. The valid IDs are:

ID	Description and Version and Release
997	X12 997 - Prior to Version 3 Release 5
997V35	X12 997 - Version 3 Release 5 or later
999	UCS 999 - All UCS
CONTRL	CONTRL - Prior to Version 2 Release 1
CONTV21	CONTRL - Version 2 Release 1 or later

Note: The functional acknowledgments generated by DataInterchange will be negative only if errors defined by the functional acknowledgment are encountered.

Mapping Your Application Data to an EDI Standard Transaction Set

In this field:

Group level FA only

Enter:

The code that indicates whether the functional acknowledgment generated should contain response segments for all transactions or only the segments for rejected transactions. For example, if generating a 997 and all transactions are accepted without error, and you only want the AK1 and AK9 segments generated, enter a Y for this code. The valid IDs are:

Value	Description
Y	Generates only the group level segments and rejected transaction segments
N	Generates both rejected and accepted segments

Inbound envelope
used on FA

The code that indicates whether you want the translator to switch and move inbound envelope data to the outbound functional acknowledgment envelope. The following fields will be moved or switched for each envelope type:

E envelope type

UNB01 to UNB01, UNB02 to UNB02, UNB03 to UNB06,
UNB04 to UNB07, UNB05 to UNB08, UNB06 to UNB03,
UNB07 to UNB04, UNB08 to UNB05, UNB15 to UNB15,
UNB17 to UNB17, UNG02 to UNG04, UNG03 to UNG05,
UNG04 to UNG02, UNG05 to UNG03, UNG09 to UNG09,
UNG10 to UNG10, UNG11 to UNG11.

I envelope type

ICS02 to ICS02, ICS03 to ICS03, ICS04 to ICS06,
ICS05 to ICS07, GS02 to GS03, GS07 to GS07,
GS08 to GS08.

T envelope type

STX01 to STX01, STX02 to STX02, STX03 to STX05,
STX04 to STX06, STX05 to STX03, STX06 to STX04,
STX12 to STX12.

U envelope type

BG02 to BG03, BG03 to BG02, BG04 to BG05,
BG05 to BG04, GS02 to GS03, GS03 to GS02,
GS07 to GS07, GS08 to GS08.

X envelope type

ISA05 to ISA07, ISA06 to ISA08, ISA07 to ISA05,
ISA08 to ISA06, ISA11 to ISA11, ISA12 to ISA12,
GS02 to GS03, GS03 to GS02, GS07 to GS07,
GS08 to GS08.

The envelope profile member name entered on the receive usage will be used if no override is provided.

Valid values are Y (yes) or N (no).

In this field:

Switch appl routing
IDs on FA

Enter:

The code that indicates whether you want the translator to switch the application routing ID on functional acknowledgments, so in groups being received:

- The application sender ID will be the application receiver ID in the generated functional acknowledgment.
- The application receiver ID will be the application sender ID in the generated functional acknowledgment.

Valid values are Y (yes) or N (no).

Entries in the FAENV file will override this specification. See “Sending Functional Acknowledgments” on page 5-48 for more information.

Activate usage

The code that indicates whether the usage you are defining is the active usage for this trading partner nickname, standard transaction, and sender or receiver combination. Valid values are Y (yes) or N (no).

This field works with the **Test usage** field to provide one active usage for production and one active usage for testing. Because only one of each (for a trading partner nickname, standard transaction, and sender or receiver combination) can be active at a time, entering Y deactivates the current usage if one of this type is active. The valid combinations are:

Active	Test	Usage
Y	Y	Choice 1 for testing
Y	N	Production and choice 2 for testing
N	Y	Not used
N	N	Not used

Usage indicator

Enter P (for production), T (for test), or I (for information) to indicate the type of usage.

The usual mode of operation for a production message is that a production usage will be used if an active production usage exists. Otherwise, an error will occur.

The usual mode of operation for a test message is that a test usage will be used if an active test usage exists. Otherwise, a production usage will be used if an active production usage exists. Otherwise, an error will occur.

The usual mode of operation for an information message is that an information usage will be used if an active information usage exists. Otherwise, a production usage will be used if an active production usage exists. Otherwise, an error will occur.

Log application data

The code that indicates whether you want the translator to save an image of this transaction in application format in the event log. Valid values are Y (yes) or N (no).

Trading Partner Usage Overrides for Receiving

The Trading Partner Usage Overrides for Receiving panel (TP27) is displayed after you enter the usage information on the Add Trading Partner Usage for Receiving panel (TP22). Use this panel to further define the trading partner usage.

Note: All the fields on this panel are optional, but if you enter a value here, it overrides the value that the translator normally uses.

TP27 Trading Partner Usage Overrides for Receiving

Transaction ID : INVOICERCVE
 Standard transaction ID : 810
 Trading partner nickname : WIDGETCOMPANY
 Application routing (sender) . . : DEPT31
 Application routing (receiver) . :

Standard envelope profile member . _____
 Application password _____
 Application file name _____
 Application file type — +
 Validation level (0/1/2) —
 Acceptable error level (0/1/2) . . —
 Data overlay check (Y/N) — +
 Unexpected data element check (Y/N) — +
 Unexpected segment check (Y/N) . . — +

The following list describes the fields of the Trading Partner Usage Overrides for Receiving panel (TP27).

In this field:	Enter:
Transaction ID	The ID of this transaction mapping. The system supplies this value for your reference.
Data format ID	The data format you are using for this transaction mapping. The system supplies this value for your reference.
Trading partner nickname	The trading partner profile member that describes this trading partner. The system supplies this value for your reference.
Application routing (sender)	The specific sender within the trading partner's company. The system supplies this value for your reference.
Application routing (receiver)	The specific receiver within your company. The system supplies this value for your reference.
Standard profile member name	<p>The envelope profile member the translator uses to build a functional acknowledgment for this trading partner transaction. It must match the name of an existing envelope profile member. An entry here overrides the value with the same name as the standard ID used in the mapping.</p> <p>You can define a generic standard profile member name by entering an ampersand (&) followed by a 1- to 6-character base name. At run time, the Envelope profile suffix from the trading partner profile is appended to the base name to dynamically generate the name of the standard profile member name to use when enveloping the functional acknowledgment.</p>
Application password	The value used to verify the password in the functional group envelope. An entry here overrides the password in the envelope profile member.

In this field:

Application file name

Enter:

For MVS, the ddname of the file in which received and translated transactions are stored or the name of a DataInterchange MQSeries Queue profile member. An entry here overrides the application file name in the data format.

For CICS, where the data is stored or how it is processed:

- Where stored:
 - A VSAM entry sequenced data set ddname
 - A temporary storage queue name
 - A transient data queue name
 - An MQSeries Queue via DataInterchange MQSeries Queue profile member
- How processed:
 - The name of a response program that gains control after the transaction is translated
 - The name of a response transaction that gains control after the transaction is translated

Application file type

For MVS, this field is either left blank if the **Application file name** field is a ddname or the value of MQ is supplied if the application file **Application file name** field specifies a DataInterchange MQSeries Queue profile member.

For CICS, this field defines the type of file indicated in the **Application file name** field. Valid values are:

Value	Description
VS	VSAM entry sequenced data set ddname
TM	Temporary storage queue (main)
TS	Temporary storage queue (auxiliary)
TD	Transient data queue
MQ	DataInterchange MQSeries Queue profile member
PG	Response program
TX	Response transaction

For details about the data your response transaction or program processes, see the *DataInterchange Programmer's Reference*.

Validation level

The level of validation during translation. Valid values are:

Value Requests

- 0 No additional validation over the mandatory validation done by the translator to convert data between the application and standard data types. With this level of validation, any validation tables specified during mapping are ignored.
- 1 Level 0 validation plus the use of the validation tables specified during mapping.
- 2 Level 1 validation plus verification that the data values being supplied are consistent with the field's data type (for example, to verify that a DT field has a valid date, or a numeric field contains numeric data only).

The default is 0. For more information about validation, see "Validation During Mapping" on page 9-37.

Mapping Your Application Data to an EDI Standard Transaction Set

In this field:	Enter:								
Acceptable error level	The code that indicates the acceptable level of errors for this trading partner transaction. Valid values are: <table><tr><th>Value</th><th>Accepts</th></tr><tr><td>0</td><td>Only transactions without errors</td></tr><tr><td>1</td><td>Transactions that have data element errors</td></tr><tr><td>2</td><td>Transactions that have segment errors</td></tr></table> The default is 0.	Value	Accepts	0	Only transactions without errors	1	Transactions that have data element errors	2	Transactions that have segment errors
Value	Accepts								
0	Only transactions without errors								
1	Transactions that have data element errors								
2	Transactions that have segment errors								
Data overlay check	The code that indicates whether you want the translator to flag errors when moving data from the standard field to an application field causes data to be overlaid. Data overlay can occur if: <ul style="list-style-type: none">• You are mapping data from a repeating standard segment to a nonrepeating application structure• More than one occurrence of a qualified segment, loop, or field is received when only one is expected Valid values are Y (yes) or N (no).								
Unexpected data element check	The code that indicates whether you want the translator to flag an error when the standard data includes unmapped data elements. Unmapped data elements can indicate that your trading partner's application changed and now includes new data elements in mapped segments. Valid values are Y (yes) or N (no).								
Unexpected segment check	The code that indicates whether you want the translator to flag an error when the standard data includes segments that are not mapped. Unmapped segments can indicate that your trading partner's application has changed and includes new segments. Valid values are Y (yes) or N (no).								

Defining Generic Receive Usages

When transactions received from multiple trading partners can use the same mapping, a generic receive usage can be defined to DataInterchange to handle multiple trading partners with a single usage and map.

Receive Usage Definition

When defining the generic receive usages, a trading partner nickname with only an ampersand (&) is a special form indicating that it is a generic usage. The generic usage is selected when the normal selection process using the trading partner nickname does not find a receive usage.

Many generic receive usages can be defined as long as one of the listed match criteria is different, (such as AS, AR, Agency, Version, or Release, production/test).

Determining the Trading Partner

The trading partner nickname is determined by searching the trading partner profile using envelope information. This trading partner nickname is then used to select a receive usage.

An indicator (Allow generic rcv) is defined in the trading partner profile to specify whether generic receive usages are allowed for this trading partner. A value of 'Y' indicates that generic receive usages are allowed.

Selecting the Receive Usage

DataInterchange uses the following steps to select either a normal usage (by trading partner nickname) or a generic usage:

1. Retrieve all normal usages using the trading partner nickname and the standard transaction set ID. The current weighting algorithm criteria (AS, AR, Agency, Version, Release, production/test) is used to find a receive usage. If a normal receive usage is found, it is used for translation.
2. If a normal receive usage is not found and generic receive usages are allowed, the trading partner nickname part of the key is changed to an ampersand (&), and the generic usages are retrieved. If generic receive usages are found, the current weighting algorithm is used to select a receive usage. If a receive usage is not found, an error occurs indicating this.

Functional Acknowledgment Processing

The envelope profile member name in the receive usage override (used for functional acknowledgment processing) can be specified as a generic envelope profile member name and will be retrieved and used to generate a functional acknowledgment. For information on using generic profile envelope profile members, see “Standard Envelope Profile Member Selection” on page 9-87. For additional information on functional acknowledgment processing, see “Requesting, Sending, and Receiving Functional Acknowledgments” on page 5-48.

Transaction Store Considerations

For tracking purposes, the actual trading partner nickname is saved in the transaction store rather than the generic nickname.

Management Reporting Considerations

For management reporting, the statistics for all transactions from all trading partners using a generic usage are collected under the usage trading partner nickname, and no statistics are collected by the actual trading partner nickname. Use a **PERFORM EXTRACT** to capture data and write your own programs to present the results.

Export/Import Considerations

When exporting generic receive usages, the trading partner profile members to be used by that usage are unknown to export and are not exported. They must be exported individually.

If a generic envelope profile member name is specified in the receive usage override, the actual envelope profile members are unknown to export and are not exported. The members must be exported individually.

Migrating a Transaction Mapping

DataInterchange’s Migration Mapping lets you copy a trading partner transaction to a different standard or standard transaction. Migration Mapping then resolves the standard differences. You can also copy the associated trading partner usages along with the transaction, or change from a test usage to a production usage, or from a production usage to a test usage.

Mapping Your Application Data to an EDI Standard Transaction Set

To migrate a transaction mapping to another transaction ID, follow these steps:

1. From the Administrator's Menu (MP01), select **Trading Partner Transactions (Mapping)**. The Transaction Mappings panel (TP01) is displayed.
2. Type **I** in action column next to the transaction mapping you want to migrate, and press Enter.

Add	Copy	Delete	List	Map	migrate	Print	Update	Generate	Where used
TP01			Transaction Mappings						1 to 8 of 8
A	Transaction ID	Send/Recv	Gen Reqd	Type	Description				
i	INVOICERCVE	N			Invoice mapping for receive				
-	INVOICERCVE2	R	Y	P	New Invoice mapping for receive				
-	INVOICESEND	S	N		Invoice mapping for send				
-	INVOICESEND2	S	Y		New Invoice mapping for send				
-	PURCORDRCVE	R	Y		Purchase order mapping for receive				
-	PURCORDSEND	S	N	C	Purchase order mapping for send				
g	SAMPLEPORECV	S	Y		Sample Purchase Order Receive				
-	SAMPLEPOSEND	S	Y		Sample Purchase Order Send				

The Migrate Transaction Mapping panel (TP36) is displayed.

For DI for MVS, the panel looks like this:

TP36	Migrate Transaction Mapping
Transaction ID	INVOICERCVE
To transaction ID	invoicercveprod
Transaction description . . .	Invoice mapping for receive
Standard ID	X12V2R3
Standard transaction ID . . .	810
In control file name	
Out control file name	
Replace mapping (Y/N)	y +
Migrate usages (Y/N)	y +

For DI for CICS, the panel looks like this:

TP36	Migrate Transaction Mapping
Transaction ID	INVOICERCVE
To transaction ID	invoicercveprod
Transaction description . . .	Invoice mapping for receive
Standard ID	X12V2R3
Standard transaction ID . . .	810
In control file name	
In control file type	-
Out control file name	
Out control file type	-
Replace mapping (Y/N)	y +
Migrate usages (Y/N)	y +

Mapping Your Application Data to an EDI Standard Transaction Set

3. Complete the fields as follows:

In this field:	Enter:
To transaction ID	The name you want to use for the new mapping transaction ID. You can use any combination of characters in the ALPHANUM table, up to 16 characters. Embedded blanks are not allowed.
Transaction description	A brief description of the new transaction mapping. You can use any character data, up to 46 characters.
Standard ID	The ID of the standard you want to migrate the transaction mapping to. If you do not enter a value, the current standard is used. You can use any combination of characters in the ALPHANUM table, up to 8 characters. Embedded blanks are not allowed.
Standard transaction ID	The ID of the standard transaction you want to migrate the transaction mapping to. If you do not enter a value, the current standard transaction is used. You can use any combination of characters in the ALPHANUM table, up to 8 characters. Embedded blanks are not allowed.
In control file name	<p>For DI for MVS, the actual file name of the input control file for migration mapping. The mapping migration input control file contains records to assist the migration service in matching old segments and new segments. If you enter a file name, that file must exist so that dynamic allocation can be used to relate the file name to a ddname.</p> <p>For DI for CICS, the name of the temporary storage queue (auxiliary storage): TM for temporary storage queue (main storage), transient data queue, or a VSAM entry sequenced data set used as the input control file.</p> <p>The maximum length of this field is 44 characters for DI for MVS, and 8 characters for DI for CICS.</p> <p>For more information, see the <i>DataInterchange Programmer's Reference</i> under INCNTL.</p>
In control file type	For DI for CICS only, the file type of the input control file: TS for temporary storage queue, TD for transient data queue, and VS for VSAM entry sequenced data set. The default is TS. The maximum length of this field is 2 characters.
Out control file name	<p>For DI for MVS, the actual file name of the output control file for migration mapping. The mapping migration control records are written to this file for you to view and understand how the migration was performed. The records can also be moved to the In control file, edited, and resubmitted during a mapping migration run to correct any problems. If you enter a file name, that file must exist so that dynamic allocation can be used to relate the file name to a ddname.</p> <p>For DI for CICS, the name of the temporary storage queue (auxiliary storage): TM for temporary storage queue (main storage), transient data queue, or a VSAM entry sequenced data set used as the output control file.</p> <p>The maximum length of this field is 44 characters for DI for MVS, and 8 characters for DI for CICS.</p> <p>For more information, see the <i>DataInterchange Programmer's Reference</i> under INCNTL.</p>

Mapping Your Application Data to an EDI Standard Transaction Set

In this field:

Out control file type

Enter:

For DI for CICS only, the file type of the input control file: TS for temporary storage queue, TD for transient data queue, and VS for VSAM entry sequenced data set. The default is TS. The maximum length of this field is 2 characters.

Replace mapping
(Y/N)

Y to replace the mapping if it already exists, or **N** if the mapping will not be replaced. If the **from** and **to** transaction IDs are the same, you must enter Y. The default is N.

Migrate usages (Y/N)

Y to migrate trading partner usages, if they exist, along with the transaction mapping, or **N** to migrate only the transaction mapping. The default is N.

If you typed **Y** in the **Migrate usages** field, and the transaction mapping you are migrating contains send usages, the Migrate Trading Partner Usage for Sending panel (TP37) is displayed.

eXclude	List	migrate
TP37	Migrate Trading Partner Usage for Sending	1 to 2 of 2
From transaction ID : INVOICESEND		
To transaction ID : INVOICESENDPROD		
Activate (Y/N) - +		
Migrate test to production (Y/N) . . - +		
Migrate production to test (Y/N) . . - +		
Ref	Trading Partner	
A Num Internal Trading Partner ID	Nickname	Active Test
- 001 123456789	GM01	Y N
- 002 456789012	GM02	Y N

4. Complete the fields as follows:

In this field:

Activate (Y/N)

Enter:

Y to make the usages you are migrating the active usage for this internal trading partner/data format combination, or **N** to make them inactive. If you specify Y, any existing active usage for this combination is made inactive. The default is N.

Migrate test to
production (Y/N)

Y to change test usages to production usages, or **N** to leave them as test usages. The default is N.

Migrate production to
test (Y/N)

Y to change production usages to test usages, or **N** to leave them as production usages. The default is N.

5. Indicate which usages you want to migrate, and press Enter.

To migrate all usages, enter an **I** on the command line.

If you typed **Y** in the **Migrate usages** field, and the transaction mapping you are migrating contains receive usages, the Migrate Trading Partner Usage for Receiving panel (TP38) is displayed.

```

eXclude List mIgrate
-----
TP38          Migrate Trading Partner Usage for Receiving          1 to 2 of 2

From transaction ID . . . . . : INVOICERCVE
To transaction ID . . . . . : INVOICERCVEPROD
Activate (Y/N) . . . . . - +
Migrate test to production (Y/N) . . - +
Migrate production to test (Y/N) . . - +

  Ref  Trading Partner
  A  Num  Nickname  Application Sender(S) or Receiver(R) ID  Active  Test
-   -   -   -
-   1  GM01          Y          N
-   2  GM02          Y          N
    
```

6. Complete the fields as follows:

In this field:

Activate (Y/N)

Enter:

Y to make the usages you are migrating the active usage for this trading partner nickname/standard transaction ID combination, or **N** to make them inactive. If you specify Y, any existing active usage for this combination is made inactive. The default is N.

Migrate test to
production (Y/N)

Y to change test usages to production usages, or **N** to leave them as test usages. The default is N.

Migrate production to
test (Y/N)

Y to change production usages to test usages, or **N** to leave them as production usages. The default is N.

7. Press Enter when you are finished indicating which usages you want to migrate.

To migrate all usages, enter an **I** on the command line.

DataInterchange migrates the transaction mapping and usages that you specified.

Migrating Trading Partner Usages

In addition to migrating entire transaction mappings, you can migrate just the trading partner usages from the Trading Partner Usage for Sending panel (TP16) or Trading Partner Usage for Receiving panel (TP21).

To migrate trading partner usages, follow these steps:

1. On the Transaction Mappings panel (TP01), type **W** in the action column next to the transaction containing the usages you want to migrate, and press Enter.

The Trading Partner Usage for Sending panel (TP16) is displayed for a send transaction, or the Trading Partner Usage for Receiving panel (TP21) is displayed for a receive transaction.

Mapping Your Application Data to an EDI Standard Transaction Set

2. Type **I** in the action column next to the usages you want to migrate, and press Enter.

To migrate all usages, enter an **I** on the command line.

```
acTivate  Add  Copy  Delete  eXclude  List  mIgrate  Print  Update
Where used

TP16          Trading Partner Usage for Sending          1 to 2 of 2

Transaction ID . . . . . : INVOICESEND
Transaction description . . : Invoice mapping for send
Data format ID . . . . . : INVOICE

  Ref          Trading Partner
  A  Num      Internal Trading Partner ID      Nickname      Active  Test
  i  1      123456789                          GM01          Y
  i  2      456789012                          GM02          Y
```

The Migrate Trading Partner Usages panel (TP39) panel is displayed.

```
TP16          Trading Partner Usage for Sending          1 to 2 of 2

Transaction ID . . . . . : INVOICESEND
Transaction description . . : Invoice mapping for send
Data format ID . . . . . : INVOICE

  Ref          Trading Partner
  A  Num      Internal Trading Partner ID      Nickname      Active  Test
  i  1
  i  2      TP39          Migrate Trading Partner Usages

      Transaction ID . . . . . : INVOICESEND
      To transaction ID . . . . .
      Activate (Y/N) . . . . . N +
      Migrate test to production . . N +
      Migrate production to test . . N +
```

3. Complete the fields as follows:

In this field:

To transaction ID

Enter:

The transaction ID you want to migrate the usages to. You can use any combination of characters in the ALPHANUM table, up to 16 characters. Embedded blanks are not allowed.

Activate (Y/N)

Y to make the usages you are migrating the active usage, or **N** to make them inactive. If you specify Y, any existing active usage for this key is made inactive. The default is N.

Migrate test to
production (Y/N)

Y to change test usages to production usages, or **N** to leave them as test usages. The default is N.

Migrate production to
test (Y/N)

Y to change production usages to test usages, or **N** to leave them as production usages. The default is N.

4. Press Enter when you are finished indicating which trading partner usages you want to migrate.

Generating Control Strings

Before your transaction mapping will work, you must generate a control string. DataInterchange uses the data you created during the mapping process, as well as data from the standards and from your ADF, as input to a program that compiles the data and generates a control string. During this process, DataInterchange checks for errors in the map you created. Minor errors are displayed on the panel. Severe errors are written to the event log.

Generating the control string is the last step after adding or updating a map or an ADF. The **Gen Reqd** field on the Transaction Mappings panel (TP01) contains a Y if the current control string no longer matches the transaction mapping or if generation is otherwise required.

To generate a control string, follow these steps:

1. On the Transaction Mappings panel (TP01) panel, type **G** in the action column next to the transaction, and press Enter.

Note: The TYPE column specifies whether the trading partner transaction was previously transferred from the host to the DataInterchange Client. If the value of the TYPE column is C, I, or P, the TPT (map) was previously moved to the DataInterchange Client. It is questionable whether the host TPT should be generated, because the most recent copy of the map source is on DataInterchange Client.

- C Indicates the host map was converted to a DataInterchange Client map database format. It also indicates the control string residing on the host was generated on DataInterchange Client.
- I Indicates the host map was converted to a DataInterchange Client Map database format, but the host control string was generated using the host map.
- P Indicates the host has only the map header; the complete map is stored in the DataInterchange Client map database. The control string, if any, was generated on the DataInterchange Client.

Completing the generation would overlay a DataInterchange Client control string. If this is the case, a warning message will be displayed informing you of this situation. Carefully determine if this is what you want to do.

Add	Copy	Delete	List	Map	migrate	Print	Update	Generate	Where used
TP01				Transaction Mappings					1 to 8 of 8
A	Transaction ID	Send/Recv	Gen Reqd	Type	Description				
—	INVOICERCVE	R	N		Invoice mapping for receive				
—	INVOICERCVE2	R	Y	P	New Invoice mapping for receive				
—	INVOICESEND	S	N		Invoice mapping for send				
—	INVOICESEND2	S	Y		New Invoice mapping for send				
—	PURCORDRCVE	R	Y		Purchase order mapping for receive				
—	PURCORDSEND	S	N	C	Purchase order mapping for send				
g	SAMPLEPORECV	S	Y		Sample Purchase Order Receive				
—	SAMPLEPOSEND	S	Y		Sample Purchase Order Send				

If the string is not generated, use the **Event logging** option on the Administrator's Menu (MP01) to view the event log for application EDIMP to see the errors generated.

Mapping Your Application Data to an EDI Standard Transaction Set

After the control string is generated, the Transaction Mappings panel (TP01) is redisplayed with a message at the bottom of the panel, telling you the control string was generated successfully.

- To print the transaction mapping, type **P** in the action column next to the transaction, and press Enter.

Add	Copy	Delete	List	Map	migrate	Print	Update	Generate	Where used
TP01			Transaction Mappings						1 to 8 of 8
A	Transaction ID	Send/Recv	Gen Req	Type	Description				
	INVOICERCVE	R	N		Invoice mapping for receive				
—	INVOICERCVE2	R	Y	P	New Invoice mapping for receive				
—	INVOICESEND	S	N		Invoice mapping for send				
—	INVOICESEND2	S	Y		New Invoice mapping for send				
—	PURCORDRCVE	R	Y		Purchase order mapping for receive				
—	PURCORDSEND	S	N	C	Purchase order mapping for send				
P	SAMPLEPORECV	S	N		Sample Purchase Order Receive				
	SAMPLEPOSEND	S	N		Sample Purchase Order Send				

DataInterchange prints the transaction mapping, then redisplay the Transaction Mappings panel (TP01). The transaction mapping printout is shown in Figure 9-8.

- Press F3 (Exit) to return to the Administrator's Menu (MP01).

MAP1		Standard-to-Application Mapping						Date: 96/04/22			
Page: 1								Time: 17:37:40			
Trading partner transaction ID :		SAMPLEPORECV				Send/receive :		R			
Description :		Sample Purchase Order Receive									
Application Control Fields		Length									
1	PONUMBER	12									
2	PODATE	8									
STANDARD		(*** -Denotes repeated elements)				APPLICATION		(* -Denotes literals)			
Standard ID :		X12V2400		Standard transaction ID :		850		Data format ID :		PURCORD10	
Loop ID	Seg/De ID	Req Des	Data Type	Length Min Max	Validation Table	Field Name/Special Handling	Data Type	Length	Edit Type	Exit/Table Name	
	BEG	M									
	324	M	AN	1 22		PONUMBER	AC	12			
	323	M	DT	8 8		PODATE	DT	8	DATE-17		
	NTE	F	Multiple occurrence : NOTE			Number of maps :	1	Maximum repeats :	100		
	3	M	AN	1 60		NOTETEXT	CH	35			
	FOB	O									
	146	M	ID	2 2	146	FOB	CH	2	VALID	146	
100 >>>	LOOP START		Qualifying data element : 66			Number of maps :	3	Maximum repeats :	200		
	N1	O									
	93	C	AN	1 35		BILLTNAME	CH	18			
	66	C	ID	1 2	66	BILLTOIDCODE	CH	2			
			Occurrence number 1			Number of maps :	1	Maximum repeats :	2		
	N3	O	AN	1 35		BILLTOADDRLN1	CH	18			
	N4	O									
	19	C	AN	2 19		BILLTOCITY	CH	11			
	156	C	ID	2 2		BILLTOSTATE	CH	2			
	116	O	ID	5 9		BILLTOZIP	CH	5			
100 >>>	LOOP END										
100 >>>	LOOP START		Qualifying data element : 66								
	N1	O									
	93	C	AN	1 35		SHIPTNAME	CH	18			
	66	C	ID	1 2	66	SHIPTOIDCODE	CH	2			
			Occurrence number 1			Number of maps :	1	Maximum repeats :	2		
	N3	O	AN	1 35		SHIPTOADDRLN1	CH	18			
	N4	O									

Figure 9-8 (Part 1 of 2). Receive Transaction Mapping Printout

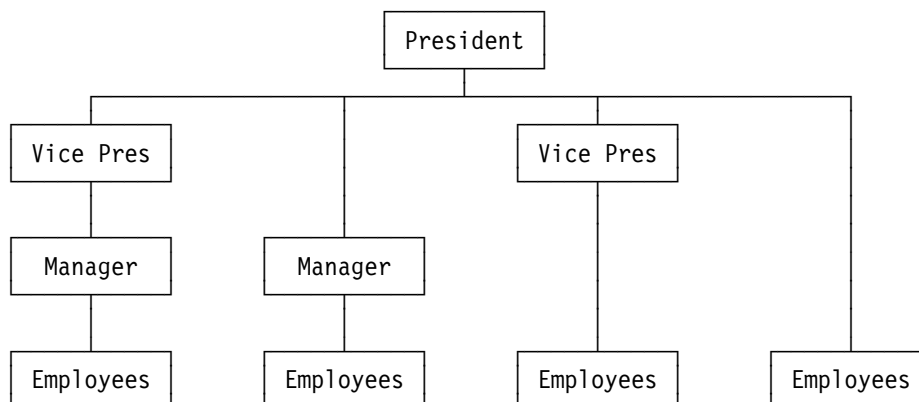
Mapping Your Application Data to an EDI Standard Transaction Set

MAP1		Standard-to-Application Mapping							Date: 96/04/22	
Page: 2									Time: 17:37:40	
Trading partner transaction ID : SAMPLEPORECV		Send/receive : R								
Description : sample Purchase Order Receive										
STANDARD (*** -Denotes repeated elements)		APPLICATION (* -Denotes literals)								
Standard ID : X12V2400		Standard transaction ID : 850							Data format ID : PURCORD10	
Loop ID	Seg/De ID	Req Des	Data Type	Length Min Max	Validation Table	Field Name/Special Handling	Data Type	Length	Edit Type	Exit/Table Name
	19	C	AN	2 19		SHIPTOCITY	CH	11		
	156	C	ID	2 2		SHIPTOSTATE	CH	2		
	116	O	ID	5 9		SHIPTOZIP	CH	5		
100 >>> LOOP END										
100 >>> LOOP START		Qualifying data element : 66								
	N1	O								
	93	C	AN	1 35		ORDERBYNAME	CH	18		
	66	C	ID	1 2	66	ORDERBYIDCODE	CH	2		
						Seg/Loop qualifier = OB				
	N3	O	Occurrence number 1			Number of maps :		1	Maximum repeats : 2	
	166	M	AN	1 35		ORDERBYADDRLN1	CH	18		
	N4	O								
	19	C	AN	2 19		ORDERBYCITY	CH	11		
	156	C	ID	2 2		ORDERBYSTATE	CH	2		
	116	O	ID	5 9		ORDERBYZIP	CH	5		
100 >>> LOOP END										
200 >>> LOOP START		Multiple occurrence : LINEITEMS				Number of maps :		1	Maximum repeats : 100000	
	P01	M								
	330	M	R	1 9		QUANTITY	R	4		
	355	M	ID	2 2	355	UNITOFMEASURE	ID	2		
						Qualifier Elem 355 = ea				
	212	C	R	1 14		UNITPRICE	R2	10		
						Qualifier Elem 355 = ea				
	234	C	AN	1 30		ITEMNUMBER	AN	13		
	PID	O	Occurrence number 1			Number of maps :		1	Maximum repeats : 200	
	352	C	AN	1 80		DESCRIPTION	CH	28		
200 >>> LOOP END										
	CTT	M								
	354	M	N0	1 6		ITEMCOUNT	R	2		

Figure 9-8 (Part 2 of 2). Receive Transaction Mapping Printout

Hierarchical Loops

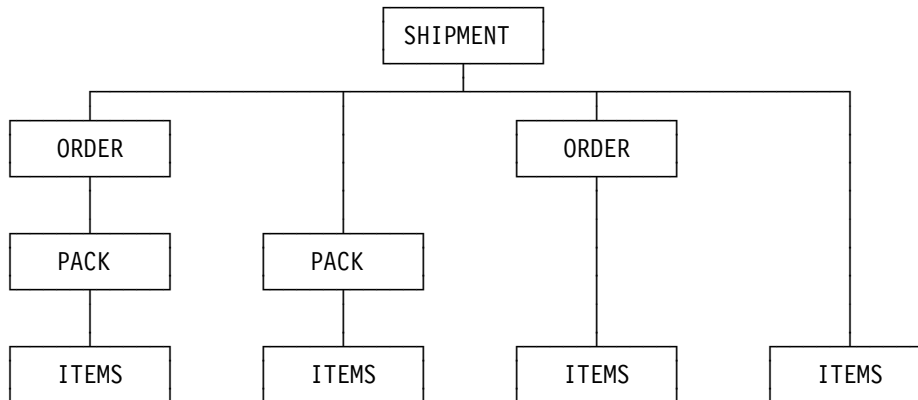
A hierarchical loop is similar to an organization chart. Just as an organization chart shows you the various groups of people and their relationship to the whole, a hierarchical loop shows you each group of data and its relationship to the whole. For example, the following figure shows an organization chart.



You can clearly see the different levels in the organization, and who reports to whom.

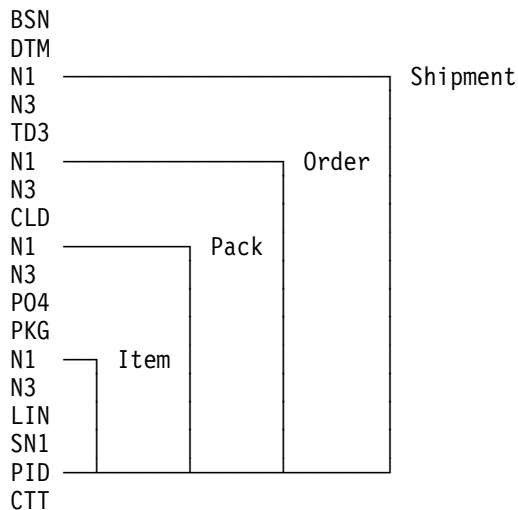
Mapping Your Application Data to an EDI Standard Transaction Set

Hierarchical loops define different levels of data, which can be used in any sequence, and skipped when appropriate, allowing you to fit the loop to your data.



The HL Segment

Sometimes you have to nest loops several levels deep to map all of your data, yet each level contains similar information, such as name and address. In the following figure, loops are nested four levels deep, and the N1 and N3 segments occur at the beginning of each loop.



The hierarchical level (HL) segment makes it easier to map these loops. Each HL segment contains information about the relationship of segments in a hierarchical loop to the other segments in the loop. This information is described in Table 9-7 on page 9-107.

Table 9-7. HL Segment

Field ID	Field Name	Description
HL01	ID number	A unique number that identifies the occurrence of the HL segment. This data element is alphanumeric and has a maximum length of 12 characters. This field usually contains a sequential number that is incremented for each occurrence of the HL segment.
HL02	Parent ID	The HL01 value of the HL segment that is the parent of the current HL segment.
HL03	Level Code	A code that indicates the level of the HL segment in the current HL loop. For example, the level code could refer to the shipment, order, or item level information in the ANSI X12 Shipping Notice transaction set.
HL04	Child code	A code that indicates if the segment has subordinate segments: 1 for subordinate segments, or 0 for no subordinate segments. The default is 0.

HL01 and HL02 provide the information for DataInterchange to determine the nesting of loops within each other.

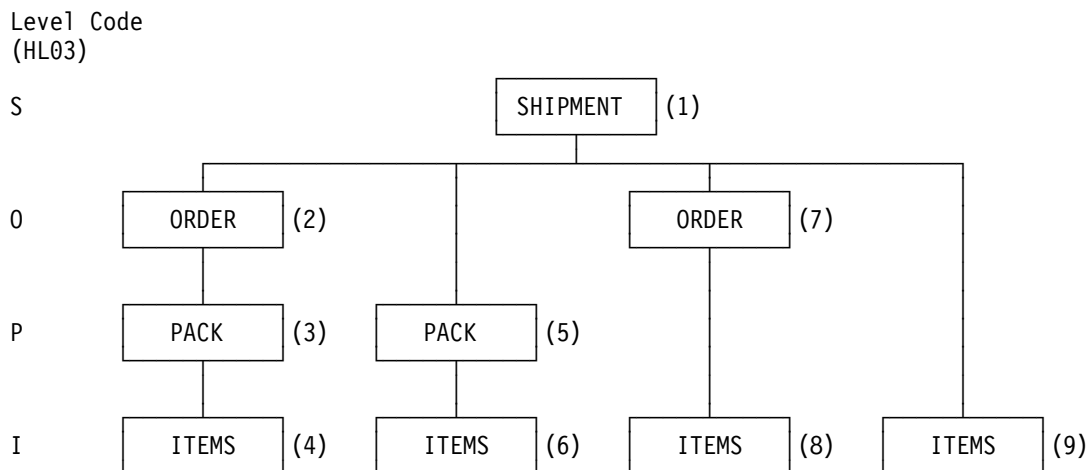
Note: The HL segment is not supported by all standards.

Preparing Hierarchical Loops

DataInterchange's HL support allows you to specify unique mapping instructions for each identifiable group of structures in a hierarchical loop. DataInterchange can handle 16 levels of nesting within the HL loop structure. To begin mapping a hierarchical level loop, follow these steps:

1. Create the hierarchy for your application data.
2. Assign node IDs to your hierarchy to uniquely identify logically grouped segments in the hierarchical loop. This ensures that DataInterchange processes your data the way you want it to be processed. The logical grouping of segments within the HL and the hierarchical relationship is defined by standards organizations or industry groups. Trading partners then agree which segments they will use and how they will be mapped.

The following diagram shows how each group of segments in your hierarchy should be numbered in a top-down, left-right order. Use this number as the node ID value.



3. Create the application data format for this hierarchical loop. The application data format and the HL mapping tell DataInterchange how to build the hierarchical loop.

Mapping Your Application Data to an EDI Standard Transaction Set

The application data can take one of three basic formats.

- The application data format can exactly match the hierarchy being created. This optimizes performance during translation because DataInterchange does minimal processing to match the data with the hierarchy.

```
SHIPBASE
  SHIPMENT
    ORDER1
      PACK1
        ITEMS1
      PACK2
        ITEMS2
    ORDER3
      ITEMS3
    ITEMS4
```

Because a structure can only occur once within a given application data format, numbers to distinguish between the different ORDER, PACK, and ITEMS structures.

- The application data format can have all the structures defined at the same level. In this case, the order of the structures passed to DataInterchange during translation helps determine how the hierarchy is built.

```
SHIPBASE
  SHIPMENT
  ORDER
  PACK
  ITEMS
```

Given the following application data:

```
SHIPMENT
ORDER
PACK
ITEMS
SHIPMENT
ITEMS
ITEMS
```

The following occurrences of the hierarchy would be built by DataInterchange during translation:

** occurrence 1 **

SHIPMENT, ORDER, PACK, ITEMS	(based on nodes 1, 2, 3, and 4)
PACK, ITEMS	(based on nodes 5 and 6)
ORDER, ITEMS	(based on nodes 7 and 8)
ITEMS	(based on node 9)

** occurrence 2 **

SHIPMENT, ITEMS	(based on nodes 1 and 9)
ITEMS	(based on node 9)

When data is passed to the translator like this, DataInterchange builds a child level of the hierarchy for each structure received in the application data between occurrences of the parent structure or structures. For example, only the ITEMS between the current SHIPMENT and the next SHIPMENT, or end of file, will be used to create this occurrence of the hierarchical loop. Any ITEMS received before the first SHIPMENT will be ignored.

- The application data can supply a structure that contains hierarchical data.

```
SHIPBASE
  HLDATA
    SHIPMENT
      ORDER
        PACK
          ITEMS
```

When data is passed to the translator like this, the fields within HLDATA define exactly how the hierarchy will be built. HLDATA must contain the value for the HL03 element, which contains the hierarchical level code. It can also contain the HL01, HL02, and HL04 elements of the HL segment, but these are not required. If values for these fields are not present in the application data, the HL special literals, which are described in “Literal Keywords for the HL Segment,” may be used.

To create our sample hierarchy, the application data being passed to DataInterchange for translation could look like:

	where HLDATA contains			
	HL01	HL02	HL03	HL04
HLDATA	1		S	1
SHIPMENT				
HLDATA	2	1	0	1
ORDER				
HLDATA	3	2	P	1
PACK				
HLDATA	4	3	I	0
ITEMS				
HLDATA	5	1	P	1
PACK				
HLDATA	6	5	I	0
ITEMS				
HLDATA	7	1	0	1
ORDER				
HLDATA	8	7	I	0
ITEMS				
HLDATA	9	1	I	0
ITEMS				

You can also use combinations of these basic formats. For example, you might want to provide an HLDATA structure for all but the last branch of the hierarchy. The application data format would be:

```
SHIPBASE
  HLDATA
    SHIPMENT
      ORDER
        PACK
          ITEMS
SHIPMENT2
  ITEMS2
```

Literal Keywords for the HL Segment

Your application data may not contain the information the translator needs to create the HL segment, but you can use the following special literals to supply the values for the HL segment.

Mapping Your Application Data to an EDI Standard Transaction Set

Keyword	Description
---------	-------------

&HLID	Supplies a sequential number for each HL segment created.
&HLPID	Supplies the HLID value for the parent of the current HL.
&HCODE	Supplies the hierarchical code associated with the current HL segment.
&HCHILD	Supplies the value 1 if the current HL segment has subordinate segments.

When you map the HL segment, you can use these literal keywords by typing them in the **Literal** field of the Map Data Element panel (TP10) when you map the data elements. For **send**, they will now be automatically mapped with their corresponding special literal when the HL loop is qualified. If different mappings for these elements are desired, the elements must be remapped.

- HL element 628 will be mapped with &HLID
- HL element 734 will be mapped with &HLPID
- HL element 735 will be mapped with &HCODE
- HL element 736 will be mapped with &HCHILD

Mapping the HL Segment

This section provides an overview of mapping the HL segment, starting from the Transaction Segments panel (TP06).

1. On the Transaction Segments panel (TP06), type **M** next to the HL segment, and press Enter.

Block	Copy	Delete mapping	Line	Map	Qualify	Repeat			
TP06			Transaction Segments				1 to 8 of 36		
Transaction ID : SHIPNOTICE									
Standard ID : X12V3R2									
Standard transaction ID . . : 856									
Mapped segments are highlighted.									
A	Ref	Segment	Seg/Loop	Req	Max	Loop	Loop		
	Num	ID	Mappings	Des	Use	ID	Repeat	Description	
—	001	BSN		M	0001	0	0	Beginning Segment for S	
—	002	NTE		F	0100	0	0	Note/Special Instructio	
—	003	DTM		0	0010	0	0	Date/Time Reference	
m	004	HL		M	0001	100000	200000	Hierarchical Level	
—	005	LIN	*****	0	0001	100000	0	Item Identification	
—	006	SN1	*****	0	0001	100000	0	Item Detail (Shipment)	
—	007	SLN	*****	0	0100	100000	0	Subline Item Detail	
—	008	PRF	*****	0	0001	100000	0	Purchase Order Referenc	

The Hierarchical Loop Support panel (TP40) is displayed if no mappings exist for this HL segment, and it is the first segment of a loop. Once a mapping option is chosen, it remains in effect until all maps for that HL segment are deleted.

When all maps for an HL have been deleted, you can again select true hierarchical mapping support or standard mapping support.

2. Type a slash (/) next to the Hierarchical mapping support option, and press Enter.

Mapping Your Application Data to an EDI Standard Transaction Set

TP06		Transaction Segments		1 to 8 of 36	
Transaction ID : SHIPNOTICE					
Standard ID : X12V3R2					
Standard transaction ID . . : 856					
Mapped segments are highlighted.					
Ref	Segment	Seg/Loop	Req	Max	Loop
A	Num	ID	Mappings	Des	Use
—	001	BSN			
—	002	NTE			
—	003	DTM			
m	004	HL			
—	005	LIN			
—	006	SN1			
—	007	SLN			
—	008	PRF			

TP40 Hierarchical Loop Support

DataInterchange has optional mapping support for Hierarchical loops. Please indicate if you want this special support by entering a slash (/) below. Standard mapping support will be assumed if nothing is entered.

☒ Hierarchical mapping support

If you are mapping a send transaction, the Qualify Hierarchical Mapping for Send panel (TP41) is displayed for you to qualify this occurrence of the HL segment.

Note: You can use the Prompt function to set the list of valid field or structure names, or look at the Application Data Format. For more information, refer to the section “Prompt (F4)” on page 2-11.

TP41		Qualify Hierarchical Mapping for Send		1 to 1 of 1	
Transaction ID : SHIPNOTICE					
Standard transaction ID : 856					
Loop ID : 001000					
Base Node ID : _____					
Node ID : _____					
Hierarchical Level Code : _____					
Parent Level Code : _____					
Field containing Hierarchical Level Code : _____ +					
Field containing Parent Level Code : _____ +					
Structure name : _____ +					
Base ID	Node ID	Level code	Parent code	Structure name	

If you are mapping a receive transaction, the Qualify Hierarchical Mapping for Receive panel (TP42) is displayed for you to qualify this occurrence of the HL segment.

Note: You can use the Prompt function to set the list of valid structure names, or look at the Application Data Format. For more information, refer to the section “Prompt (F4)” on page 2-11.

Mapping Your Application Data to an EDI Standard Transaction Set

TP42	Qualify Hierarchical Mapping for Receive	1 to 1 of 1
Transaction ID	SHIPNOTICE	
Standard transaction ID . . .	856	
Loop ID	001000	
Base Node ID	_____	
Node ID.	_____	
Hierarchical Level Code . . .	_____	
Parent Level Code	_____	
Structure name	_____ +	
Base ID	Node ID	Level code
Parent code	Structure name	

The panels are the same except for two fields on the Qualify Hierarchical Mapping for Send panel (TP41): **Field containing Hierarchical Level Code** and **Field containing Parent Level Code**.

The **Transaction ID**, **Standard Transaction ID**, and **Loop ID** fields display what transaction you are mapping to which transaction set and to which loop.

The column headings at the bottom of the panel will list the mappings for the HL segment.

3. Complete the rest of the fields as follows:

In this field:

Base Node ID

Enter:

The ID of the first node (in our example, **1**). This field identifies the mappings for a complete hierarchical structure, or for a substructure within the hierarchy. This field is used to associate all the mappings for a particular occurrence.

This field is numeric and has a maximum length of 5 digits.

Node ID

The ID of the node you are mapping (in our example, **1**). This field identifies the different mappings for a hierarchical loop, and defines the order of the processing instructions for the control string generator. HL segments define a top-down/left-right ordered structure. If each node in this hierarchy is numbered in a top-down/left-right sequence, the instructions generated match the order of data that is being received.

This field is numeric and has a maximum length of 5 digits.

Hierarchical Level
Code

The hierarchical level code for this mapping (in our example, **S**). During translation, the value in the HL03 field of the HL segment is compared to this value to determine which map to use.

You must enter a value in this field.

In this field:

Parent Level Code

Enter:

The hierarchical level code of the node to which this node is subordinate, if any. For the first occurrence of the segment, leave this field blank.

If a hierarchical level code can appear in multiple places in the hierarchy, and you want the mapping to be dependent on its position within the hierarchy, provide a value here. For example, ITEMS can be the child of PACK at node 3, PACK at node 5, ORDER at node 7, or SHIPMENT at node 1. If you were qualifying ITEMS at node 6, you would type **P** to indicate that the parent of this node (PACK at node 5) is at level P.

For Send Mapping

A parent level code of zero can be used to create a generic mapping that allows you to do a single mapping for a structure even if it has several parents. For example, in our application data format, there is:

```
SHIPMENT
ORDER
PACK
ITEMS
```

You only want to map ITEMS once, even though it has three possible parents; you map it once with a parent level code of 0 (generic), then define it three times:

- Parent level code P and structure name &H
- Parent level code O and structure name &H
- Parent level code S and structure name &H

If you enter a value, this mapping is executed only when both of the following occur:

- The HL03 value of the current HL segment matches the value in the **Hierarchical Level Code** field.
- The HL03 value of the parent HL segment matches the value of this field.

For Receive Mapping

A parent level code of blank can be used to create a generic mapping that allows you to do a single mapping for all occurrences of the hierarchical level code. This allows you to create one mapping even if the hierarchical level code has multiple parent level codes.

Field containing
Hierarchical Level
Code

The name of the application data format field that contains the hierarchical level code. This field does not appear on the Qualify Hierarchical Mapping for Receive panel (TP42).

If you provide a field name, a new HL loop occurrence is created each time the structure containing the field is received. Use this field only for the base node of the hierarchy, which has no parent level code. You can specify structures for other nodes of the hierarchy, but they will be ignored because only the structure containing this field will cause a new occurrence of the HL loop to be created.

Mapping Your Application Data to an EDI Standard Transaction Set

In this field:

Field containing
Parent Level Code

Enter:

The name of the application data format field that contains the parent level code. This field does not appear on the Qualify Hierarchical Mapping for Receive panel (TP42).

If you provide a field name, the field must be in the same structure as the field containing the hierarchical level code. Use this field only for the base node of the hierarchy, which has no parent level code.

Structure name

The name of the repeating structure in the application data format that provides all data for this HL loop. If this is an inner loop definition, the structure should be within the domain of the outer loop.

For a receive transaction, if you specify a structure, this mapping always creates a new occurrence of the structure rather than contributing to a structure created by another loop mapping. The structure name can be any combination of characters in the ALPHANUM table, up to 16 characters.

If you specified a name in the **Field containing Hierarchical Level Code** field on the Qualify Hierarchical Mapping for Send panel (TP41), leave this field blank.

For send mapping, you can use the &H structure name to define a parent-child relationship for which there are no specific mapping instructions. If you use &H, a generic mapping, which has the following, will be used:

- The **Base Node ID** is blank or the value is equal to the value in the current mapping.
- The **Hierarchical Level Code** is equal to the value in the current mapping.
- The **Parent Level Code** is 0.

In this field:

Enter:

DataInterchange provides for the fact that there may be more than one application structure that indicates a hierarchical level should be created. For example, assume that 99 percent of the time, the SHIPMENT structure indicates that the SHIPMENT hierarchical level (S) should be created. However, sometimes the application data is such that the very last SHIPMENT is signaled with a structure called LASTSHIPMENT. Perhaps LASTSHIPMENT is different from SHIPMENT in that it contains some total information relative to the entire transaction, but in all other respects is identical to SHIPMENT. If this were the case, then NODE 1 of the hierarchy could be mapped twice, once with a structure name of SHIPMENT and once with a structure name of LASTSHIPMENT.

If you have multiple mappings for the same NODE ID that only differ in the structure name provided, the results of the translation will differ based on the relationships the structures have with one another. In this example, where two mappings are provided, one for SHIPMENT and one for LASTSHIPMENT:

- If no relationship exists between SHIPMENT and LASTSHIPMENT, DataInterchange will create a SHIPMENT level (S) occurrence for each SHIPMENT or LASTSHIPMENT structure that is provided in the application data.
- If a relationship exists between SHIPMENT and LASTSHIPMENT, then DataInterchange will create a SHIPMENT level (S) occurrence for each SHIPMENT structure in the application data.

For each level (S) occurrence that is successfully created using the SHIPMENT structure, DataInterchange will attempt to create a level (S) occurrence for each LASTSHIPMENT structure within the current SHIPMENT structure.

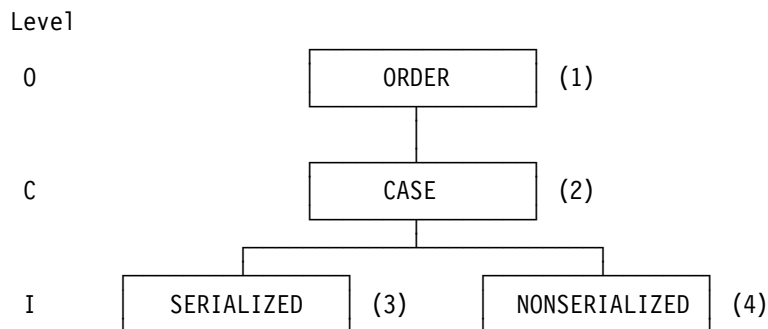
4. After you complete the fields, press Enter. DataInterchange displays the Loop Segments panel (TP14). You can now map the segments of this loop as you would any other segments.
5. When you return to the Transaction Segments panel (TP06), use the **Repeat** action to map the next occurrence of the HL segment.

After mapping all of your nodes, the following is an example of what would appear at the bottom of the Qualify Hierarchical Mapping for Send panel (TP41) or the Qualify Hierarchical Mapping for Receive panel (TP42).

Base ID	Node ID	Level code	Parent code	Structure name
1	1	S		STRUCTURE1
1	2	O	S	STRUCTURE2
1	3	P	O	STRUCTURE3
1	4	I	P	STRUCTURE4
1	5	P	S	STRUCTURE5

Hierarchical Loop Case - Ship Notice (856)

The ship notice application data contains orders. These orders consist of a number of cases. Each case contains serialized and nonserialized line items. This sample definition for the 856 has three levels.



Orders and cases have specific record layouts and identifiers. The record layouts for the serialized and nonserialized items are the same, but they are identified by different record IDs. Serialized and nonserialized items are at the same hierarchical level but are mapped differently. The trading partners agree to the following application data format for serialized and nonserialized items:

```

ORDERSTR
CASESTR
SERSTR
NOSERSTR
  
```

Following is the final qualification panel required for this mapping of the hierarchy.

TP41		Qualify Hierarchical Mapping for Send		1 to 4 of 4	
Transaction ID	:	HLEXAMPLE1		
Standard transaction ID	:	856		
Loop ID	:	001000		
Base Node ID	:	_____		
Node ID	:	_____		
Hierarchical Level Code	:	_____		
Parent Level Code	:	_____		
Field containing Hierarchical Level Code	:	_____	+	
Field containing Parent Level Code	:	_____	+	
Structure name	:	_____	+	
Base ID	Node ID	Level code	Parent code	Structure name	
1	1	O		ORDERSTR	
1	2	C	O	CASESTR	
1	3	I	C	SERSTR	
1	4	I	C	NOSERSTR	

These qualifications would cause the following to occur:

1. Each ORDERSTR received would cause Level O to be created.
2. For each Level O created, the CASESTRs associated with the current ORDERSTR will cause a Level C to be created.

Note: If there is no direct association between ORDERSTR and CASESTR defined in the ADF, then all CASESTRs between the current ORDERSTR and the next ORDERSTR or end of data will be considered to be associated with the ORDERSTR.

3. The creation of Level I occurrences depends on the relationship between SERSTRs and NOSERSTRs.

- No relationship between SERSTR and NOSERSTR

For each Level C created, each SERSTR or NOSERSTR associated with the current CASESTR will cause a Level I to be created. These will be created in the order that the structures are received.

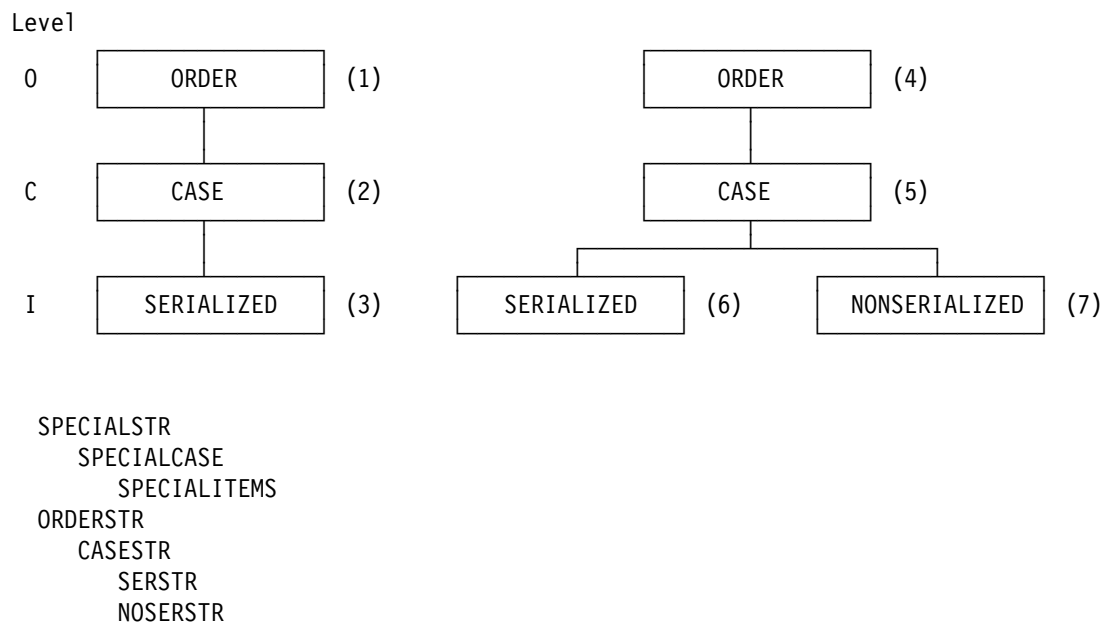
- NOSERSTR defined as part of SERSTR
 - For each Level C created, the SERSTRs associated with the current CASESTR will cause a Level I to be created.
 - Each NOSERSTR associated with the current SERSTR will cause a Level I to be created.

Hierarchical Loop Case - Ship Notice (856) with a Special Requirement

Suppose the Ship Notice (856) has the following special requirement:

The first order contains only serialized items and one case, and uses a different structure other than the rest of the orders.

The following diagram shows the hierarchy and application data format for this situation.



Mapping Your Application Data to an EDI Standard Transaction Set

The following diagram shows the final qualification panel required for this mapping of the hierarchy.

TP41		Qualify Hierarchical Mapping for Send		1 to 7 of 7		
Transaction ID	:	HLEXAMPLE1T			
Standard transaction ID	:	856			
Loop ID	:	001000			
Base Node ID	:	_____			
Node ID	:	_____			
Hierarchical Level Code	:	_____			
Parent Level Code	:	_____			
Field containing Hierarchical Level Code	:	_____ +			
Field containing Parent Level Code	:	_____ +			
Structure name	:	_____ +			
Base ID	Node ID	Level	code	Parent	code	Structure name
1	1	0				SPECIALSTR
1	2	C		0		SPECIALCASE
1	3	I		C		SPECIALITEMS
4	4	0				ORDERSTR
4	5	C		0		CASESTR
4	6	I		C		SERSTR
4	7	I		C		NOSERSTR

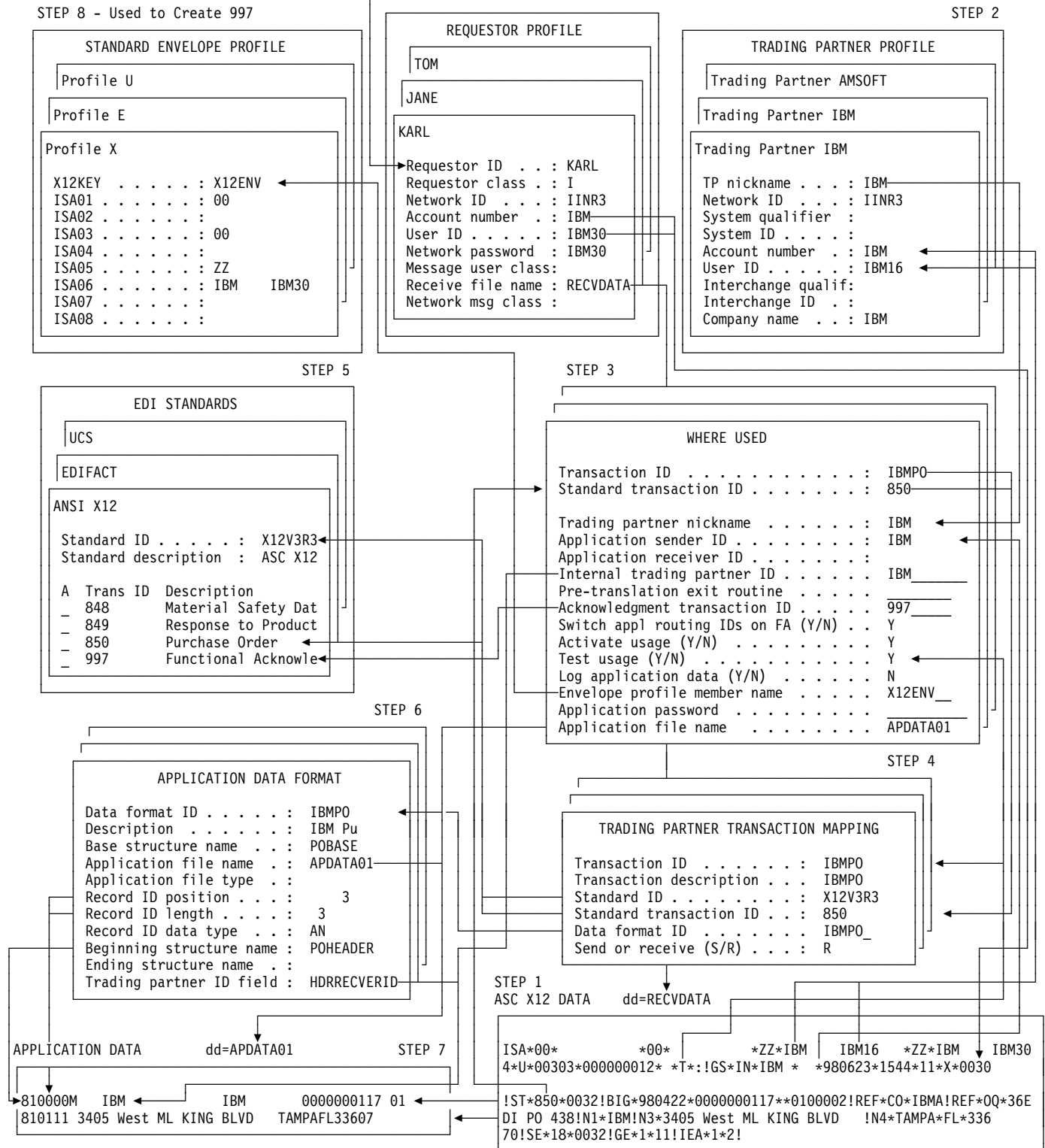
DataInterchange Inbound and Outbound Processing Flowcharts

The flowcharts on the following two pages show the processing of DataInterchange for inbound and outbound transactions. Although not all the possible sources of information are shown, most all of the commonly used sources are shown. These flowcharts can be very helpful when testing a new mapping. They help you follow the logical relationships within the product.

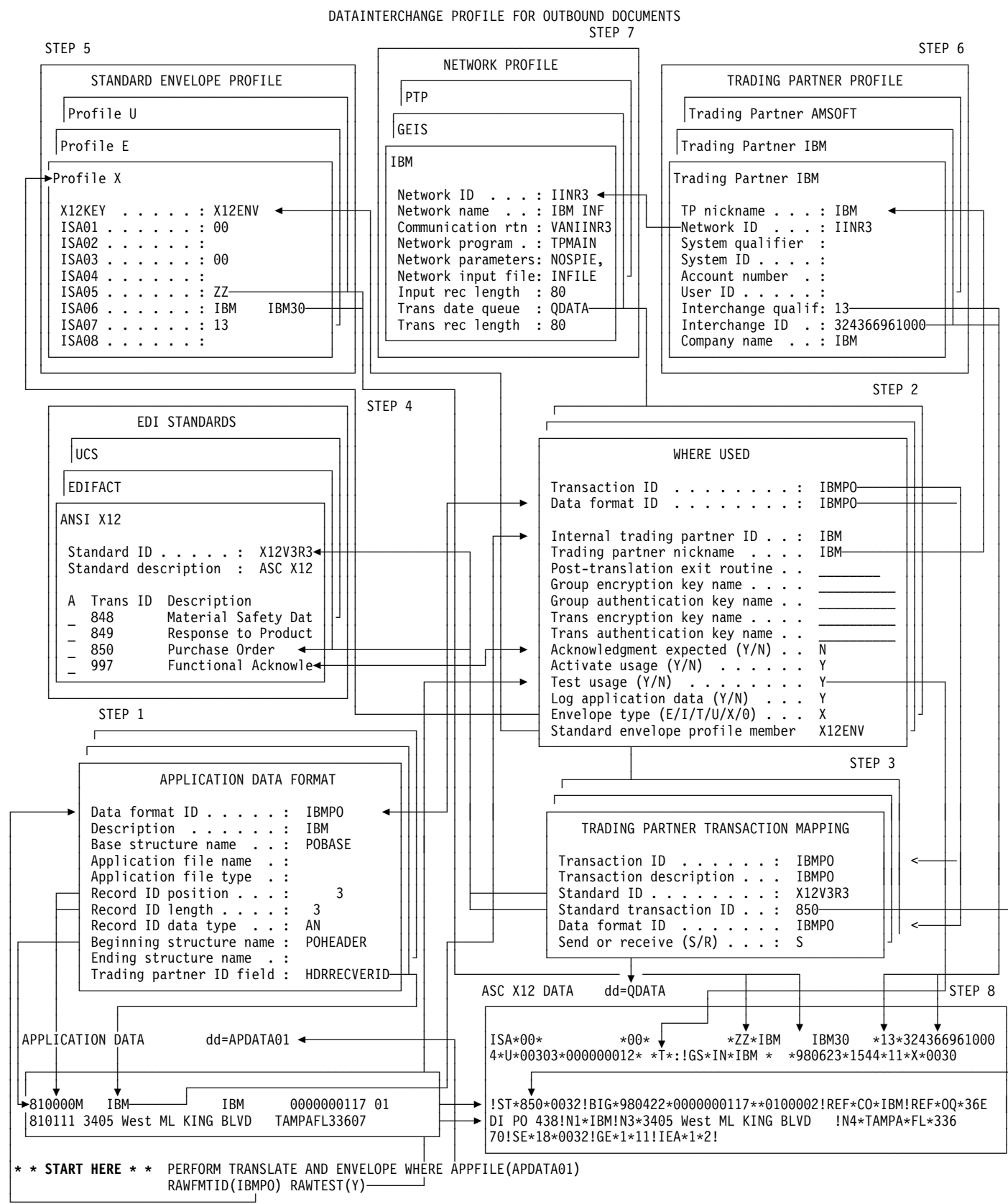
Mapping Your Application Data to an EDI Standard Transaction Set

DATAINTERCHANGE PROFILE FOR INBOUND DOCUMENTS

**** START HERE **** PERFORM RECEIVE AND TRANSLATE
WHERE REQID(KARL) RAWDATA(Y)



Mapping Your Application Data to an EDI Standard Transaction Set



Chapter 10. Managing Your EDI Data Using the Transaction Store Facility

Data Flow through the Transaction Store	10-2
Transaction Store Facility Menu (TF01)	10-6
Selecting EDI Transactions You Want to Work With	10-7
Default Criteria	10-8
Criteria Selections Panel (TF02)	10-8
Transaction Criteria	10-9
Trading Partner and Network Criteria	10-11
Envelope Criteria	10-12
Application Criteria	10-13
Store Status Criteria	10-14
Additional Selection Criteria Panel (TF08)	10-15
Sending and Receiving Transactions	10-15
Enveloping Transactions	10-15
Enveloping and Sending Transactions	10-16
Receiving Transactions	10-17
Translating Received Transactions	10-19
Managing Transactions	10-22
Reporting Transaction Status	10-23
Updating Store Status	10-24
Updating Network Status	10-25
Reporting Interchange or Group Status	10-25
Updating Interchange or Group Status	10-27
Transaction Status Codes	10-28
Panel Actions	10-31
Acknowledgment	10-31
Detail	10-31
Envelope	10-32
eVentlog	10-33
eXclude	10-33
grOups	10-33
Hold	10-33
Image	10-34
Line	10-34
Options	10-34
Print	10-35
purGe	10-36
Release	10-36
Report	10-36
Send	10-36
Summary	10-37
Transactions	10-37
Translate	10-37
Unpurge	10-37

Chapter 10. Managing Your EDI Data Using the Transaction Store Facility

This chapter describes how to use the Transaction Store Facility to manage your Electronic Data Interchange (EDI) data.

The *Transaction Store* is a collection of electronic data interchange transaction images and the control information needed to track the progress of transactions. Transactions enter the Transaction Store when they are translated for sending or when they are deenveloped after being received. All images are in standard format, without envelope header and trailer segments. Encrypted transactions that you send are not encrypted until they leave the Transaction Store for enveloping. Encrypted transactions you receive are decrypted before they enter the Transaction Store.

The Transaction Store Facility is an interactive interface to the transaction store services. With these services, you can:

- Envelope transactions
- Send transactions
- Receive and deenvelope transactions
- Translate received transactions to application format
- View or print transaction status information and functional acknowledgments
- View or print envelope status information and functional acknowledgments
- Retrieve acknowledgments from the network to update the network status of transactions
- Update the Transaction Store status by transaction, functional group, or interchange envelope.

An alternative to the Transaction Store Facility is the DataInterchange Utility, a command-level interface to DataInterchange services.

Each method has its advantages. The utility has a more extensive set of services. For example, you can use the utility to translate data from application to standard format. However, the facility lets you view status information and images on your terminal. For more information about the DataInterchange Utility, see the *DataInterchange Programmer's Reference*.

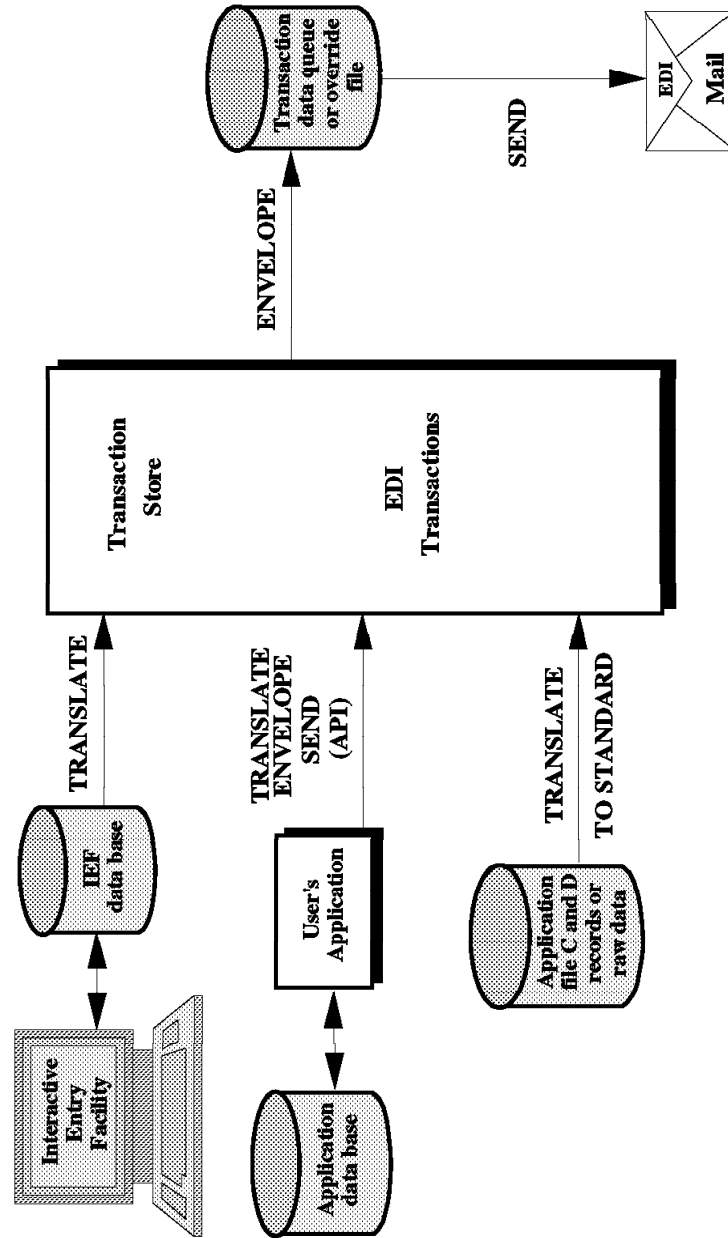
You can also work with transactions in the Transaction Store using the Interactive Entry Facility (IEF) or an application program interface. For information on using the IEF, see Chapter 13, "Using the Interactive Entry Facility (IEF)." For information on using an application program interface, see the *DataInterchange Programmer's Reference*.

Data Flow through the Transaction Store

Figure 10-1 on page 10-3 illustrates how data flows through the Transaction Store for a send transaction. Transactions enter the Transaction Store when they are translated for sending. Transactions remain in the Transaction Store until they are explicitly removed with the DataInterchange Utility's REMOVE command. When transactions are enveloped, they are copied from the Transaction Store to a transaction data queue or override file. The SEND command sends the transactions in the transaction data queue to the network.

The settings defined in the APPDEFS profile will affect the saving of images in the transaction store. For more information, see "Application Definition Profile (APPDEFS)" on page 4-11.

SEND TRANSACTION STORE DATA FLOW



DEV1165

Figure 10-1. Data Flow through Transaction Store for Sending

Managing Your Data Using the Transaction Store Facility

Figure 10-2 on page 10-5 illustrates how data flows through the Transaction Store for a receive transaction. Transactions are retrieved from the network and placed in a receive or override file. Transactions enter the Transaction Store when they are deenveloped. When transactions are translated, they are copied from the Transaction Store to an application file, IEF database, or a report file.

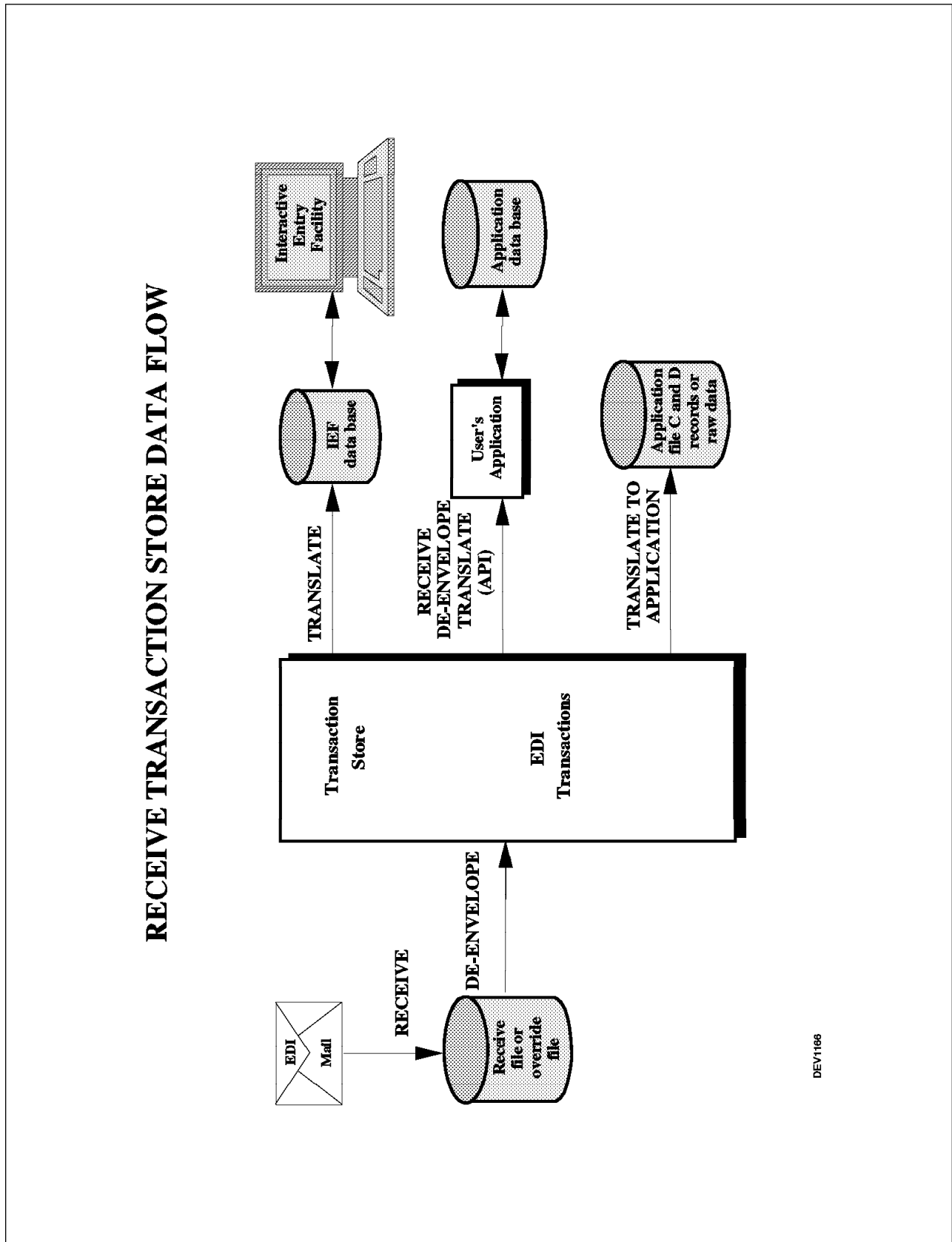


Figure 10-2. Data Flow through Transaction Store for Receiving

Transaction Store Facility Menu (TF01)

When you select *Transaction Store Facility* from the Administrator's Menu (MP01), the Transaction Store Facility Menu (TF01) is displayed.

TF01

Transaction Store Facility

Type the number of your choice and press Enter, or press the Exit key to exit.

Choice ==>

1. Envelope transactions

2. Envelope and send transactions

3. Receive and deenvelope transactions

4. Translate received transactions

5. Reenvelope transactions

6. Reenvelope and send transactions

7. Re-translate received transactions

8. Report transaction status

9. Update store status

10. Update network status

11. Report interchange/group status

12. Update interchange/group status

Table 10-1 describes the Transaction Store Facility menu options.

Table 10-1 (Page 1 of 2). Transaction Store Facility Menu (TF01) Options

Option	Description
Envelope transactions	Envelopes transactions in the Transaction Store and places them in a file ready to be sent to the network.
Envelope and send transactions	Envelopes transactions in the Transaction Store, then sends them to the network.
Receive and deenvelope transactions	Receives transactions from a network or a file, then deenvelopes and places them in the Transaction Store.
Translate received transactions	Translates received transactions to application format, then delivers to a file or IEF.
Reenvelope transactions	Envelopes transactions that have been previously enveloped or were part of a failed enveloping request. The new envelope does not affect existing envelopes.
Reenvelope and send transactions	Envelopes and sends transactions that have been previously enveloped, were part of a failed enveloping request, or that have been enveloped and sent.
Re-translate received transactions	Translates received transactions that have been previously translated.
Report transaction status	Lets you view or print transaction status information, transaction images, transaction acknowledgments, and event log entries.
Update store status	Changes the store status of a transaction in the Transaction Store. Store status can be active, held, purge pending (store time expired), and purge pending (user requested). Store status is not related to transaction or network status.

Table 10-1 (Page 2 of 2). Transaction Store Facility Menu (TF01) Options

Option	Description
Update network status	Retrieves network acknowledgments from the network. Because this task runs in the background, expect some delay before the status is updated.
Report interchange/group status	Lets you view or print interchange envelope details or images, or functional group details or images.
Update interchange/group status	Allows you to update the store status of transactions in interchange envelopes and functional groups.

Selecting EDI Transactions You Want to Work With

For some Transaction Store Facility tasks, you might want to select all available transactions. For other tasks, you might want to select only a limited set of transactions, such as the transactions received from a specific trading partner. The Criteria Selections panel (TF02) lets you define criteria for limiting the selection of transactions.

The selection criteria falls into five categories:

1. Transaction criteria

- Direction
- EDI standard transaction ID
- Transaction status
- Date and time added to the Transaction Store
- Transaction handle
- Translation error level
- Date, time, and status of functional acknowledgments
- Purge date
- Test or production usage
- Duplicate transactions

2. Trading partner and network criteria

- Trading partner nickname
- Internal trading partner ID
- Date and time sent
- Network status
- Date, time, and status of network acknowledgments

3. Envelope criteria

- Envelope type
- Interchange, functional group, and transaction control numbers
- Interchange sender or receiver
- Application sender or receiver
- Date and time enveloped

4. Application criteria

- Application ID
- Data format ID
- Application control number
- Batch ID
- Date and time delivered to the application

Managing Your Data Using the Transaction Store Facility

5. Store status criteria

Lets you select transactions by their status in the Transaction Store: Active, Held, Purge — store time expired, or Purge — user requested.

Because of the many criteria to choose from, and because some apply to one task but not another, you can have difficulty in the beginning obtaining the results you want. Try using the default selection criteria first to see the results.

Default Criteria

The following table lists the default criteria for the Transaction Store Facility Menu (TF01) options. These criteria are used if you do not specify any other criteria. A dash (–) indicates no default criteria exists for that option.

Table 10-2. Default Criteria for Transaction Store Facility Menu (TF01) Options

Menu Option	Direction	Transaction Status	Store Status
Envelope transactions	Send	Send translated (21)	Active
Envelope and send transactions	Send	Send translated (21)	Active
Receive and develope transactions	–	–	–
Translate received transactions	Receive	Received (70)	Active
Reenvelope transactions	Send	Envelope error (31)	Active
Reenvelope and send transactions	Send	Envelope error (31) through Not sent--network error (43)	Active
Re-translate received transactions	Receive	Receive translate error (73)	Active
Report transaction status	–	–	–
Update store status	–	–	–
Update network status	–	–	–
Report interchange/group status	–	–	–
Update interchange/group status	–	–	–

Criteria Selections Panel (TF02)

The Criteria Selections panel (TF02) is displayed when you select any option from the Transaction Store Facility Menu (TF01) except the *Receive and develope transactions* or *Update network status* option. This panel lists the selection criteria categories that apply to the option you selected on the Transaction Store Facility Menu (TF01). For example, if you selected *Envelope transactions*, the Criteria Selections panel (TF02) displays only the transaction criteria, trading partner and network criteria, and the application criteria categories. If you selected *Update interchange/group status*, the Criteria Selections panel (TF02) displays all five of the selection criteria categories.

Type a slash (/) next to each category you want to use, and press Enter.

TF02

Criteria Selections

The list of transactions you will work with depends on the selection criteria you enter. To see transactions based on default criteria for the task you chose, press Enter. To limit the list with specific criteria, type a slash (/) beside one or more categories and press Enter.

```

/ Transaction criteria
/ Trading partner and network criteria
/ Envelope criteria
/ Application criteria
/ Store status criteria
    
```

DataInterchange displays a criteria panel for each category you select. These panels are described in the following sections:

- “Transaction Criteria”
- “Trading Partner and Network Criteria” on page 10-11
- “Envelope Criteria” on page 10-12
- “Application Criteria” on page 10-13
- “Store Status Criteria” on page 10-14

When you specify selection criteria, DataInterchange retrieves only those transactions that meet *all* of the specified conditions.

Some selection criteria lets you specify a range. Use the first field for the low end of the range, and the second field for the high end. To specify a single value, use the first field and leave the second field blank.

For selection criteria specifying a date or time, the date and time formats must match the *Date mask* and *Time mask* fields in the language profile described in “Language Profile (LANGPROF)” on page 4-3.

Transaction Criteria

TF02

Criteria Selections

The list of transactions you will work with depends on the selection

crit
task
type

```

TF03                                Transaction Criteria
Direction (S/R) . . . . . _
Standard transaction ID . . . _
Transaction status . . . . . _ to _
Date added to store . . . . . _ to _
Time added to store . . . . . _ to _
Transaction handle . . . . . _ to _

Translation error level . . . _
Functional ack pending (Y/N) . _
Functional ack date . . . . . _ to _
Functional ack time . . . . . _ to _
Purge date . . . . . _ to _
Usage indicator (P/T/I) . . . _
Duplicate transactions (Y/N) . _
    
```

Depending on the option you selected from the Transaction Store Facility Menu (TF01), some of the fields will contain default values. Fields that do not apply to the option are not displayed.

Managing Your Data Using the Transaction Store Facility

To specify Transaction criteria, complete one or more of the following fields, and press Enter.

In this field:

Direction

Enter:

S for a send transaction, or **R** for a receive transaction.

Standard

The ID of an EDI standard transaction set, such as CREADV for a EDIFACT credit advice message, or 850 for an X12 purchase order transaction.

transaction ID

Transaction status

A 2-digit transaction status code. For send transactions, valid codes are:

Code	Status	Code	Status
20	Send translate error	50	Accepted by network
21	Send translated	51	Delivered by network
29	Trx detached - send	52	Purged by network
30	Enveloped	53	Recall requested
31	Envelope error	54	Recall request error
41	Sent with errors	55	Recalled
42	Send request error	61	Transaction accepted
43	Not sent--network error	62	Transaction rejected
46	Send started	63	Transaction accepted with errors
48	Send requested		
49	Sent to network		

For received transactions, valid codes are:

Code	Status	Code	Status
70	Received	73	Receive translate error
71	Received syntax error	74	Trx detached - recv
72	Receive translated		

These codes are described in "Transaction Status Codes" on page 10-28.

Date added to store

A date or range of dates to select receive transactions by the date they were received from the network or to select send transactions by the date they were translated for sending.

Time added to store

A time or range of times to select receive transactions by the time they were received from the network or to select send transactions by the time they were translated for sending.

Transaction handle

The ID the Transaction Store assigned to a transaction. The transaction handle is a 20-byte character field that uniquely identifies each transaction. It is the key around which DataInterchange database tables are built. This ID is a concatenation of a date, time, and a sequence number, as follows:

yyyymmddhhmmssnnnnnn

For send transactions, the date and time indicate when the transactions were translated to the standard. For receive transactions, the date and time indicate when the transactions were deenveloped.

Because it is difficult to know the exact transaction handle, you can type just the date, or date and time. The system pads the FROM value with 0s and the TO value with 9s, and retrieves all transactions whose ID falls in that range.

Translation error level

0 to select transactions with no errors, **1** for transactions with data element errors, **2** for transactions with data element and segment errors, or **3** for transactions with severe errors.

In this field:

Functional acknowledgment pending

Enter:

Y for transactions with functional acknowledgments pending, **N** for transactions with no functional acknowledgments pending.

Functional ack date

A date or range of dates to select transactions by when the last functional acknowledgment was received and deenveloped.

Functional ack time

A time or range of times to select transactions by when the last functional acknowledgment was received and deenveloped.

Purge date

A date or range of dates to select transactions by when they are due to be purged because their store time has expired.

Usage indicator

P for production transactions, **T** for test transactions, or **I** for information transactions. Leave this field blank to select transactions without regard to usage.

Duplicate transactions

Y for transactions that were part of a duplicate envelope, or **N** for transactions that were not.

Trading Partner and Network Criteria

TF02 Criteria Selections

The list of transactions you will work with depends on the selection criteria you enter. To see ALL transactions that are available for the task you chose, press Enter. To limit the list with specific criteria, type a slash (/) beside one or more categories and press Enter.

Tr	En	Ap	Tr	TF04	Trading Partner and Network Criteria
7					Trading partner nickname _____ +
-					Internal trading partner ID _____
-					Network ID _____ +
					Date sent _____ to _____
					Time sent _____ to _____
					Network status _____ +
					Network ack date _____ to _____
					Network ack time _____ to _____
					Network ack pending (Y/N) . _____ +

Depending on the option you selected from the Transaction Store Facility Menu (TF01), some of the fields will contain default values. Fields that do not apply to the option are not displayed.

To specify Trading partner and network criteria, complete one or more of the following fields, and press Enter.

In this field:

Trading partner nickname

Enter:

The nickname of a trading partner identified in the trading partner profile.

Internal trading partner ID

The internal ID of a trading partner identified in the trading partner usage for a transaction.

Network ID

A network ID defined in the network profile.

Date sent

A date or range of dates to select transactions by the date of the send request.

Managing Your Data Using the Transaction Store Facility

In this field:

Time sent

Network status

Enter:

A time or range of times to select transactions by the time of the send request.

One of the following codes to select outbound transactions by network status:

Code	Status	Code	Status
30	Enveloped	50	Accepted by network
31	Envelope error	51	Delivered by network
41	Sent with errors	52	Purged by network
42	Send request error	53	Recall requested
43	Not sent--network error	54	Recall request error
46	Send started	55	Recalled
48	Send requested		
49	Sent to network		

These codes are described in "Transaction Status Codes" on page 10-28.

Network ack date

A date or range of dates to select transactions by when the last network acknowledgment was received.

Network ack time

A time or range of times to select transactions by when the last network acknowledgment was received.

Network ack pending

Y for transactions with network acknowledgments pending, or **N** for transactions with no network acknowledgments pending.

Envelope Criteria

TF02

Criteria Selections

The list of transactions you will work with depends on the selection criteria you enter. To see ALL transactions that are available for the task you chose, press Enter. To limit the list with specific criteria, type a slash (/) beside one or more categories and press Enter.

- Tr

- Tr

7 En

- Ap

- Tr

TF05

Envelope Criteria

Envelope type -

Interchange control . . _____ to _____

Group control _____ to _____

Transaction control . . _____ to _____

Interchange sender . . _____

Interchange receiver . . _____

Application sender _____

Application receiver . . _____

Date enveloped _____ to _____

Time enveloped _____ to _____

Depending on the option you selected from the Transaction Store Facility Menu (TF01), some of the fields will contain default values. Fields that do not apply to the option are not displayed.

To specify Envelope criteria, complete one or more of the following fields, and press Enter.

In this field:
Envelope type

Enter:
A code for the type of envelope used:

Code	Envelope
E	EDIFACT
I	Interchange Control Segments (ICS)
T	Trade Data Interchange (UNTDI)
U	Uniform Communication Standard (UCS)
X	X12
0	No interchange envelope

Interchange control	The interchange control number of the transactions you want to select. If you entered 0 in the <i>Envelope type</i> field, the interchange control number is the same as the group control number.
Group control	The functional group control number of the transactions you want to select. This field applies only to transactions enveloped as part of a functional group.
Transaction control	The transaction set control number of the transactions you want to select.
Interchange sender	An interchange sender ID.
Interchange receiver	An interchange receiver ID.
Application sender	An application sender ID.
Application receiver	An application receiver ID.
Date enveloped	A date or range of dates to select transactions by when they were enveloped.
Time enveloped	A time or range of times to select transactions by when they were enveloped.

Application Criteria

TF02 Criteria Selections

The list of transactions you will work with depends on the selection criteria you enter. To see ALL transactions that are available for the task you chose, press Enter. To limit the list with specific criteria, type a slash (/) beside one or more categories and press Enter.

TF06 Application Criteria

Application ID _____

Data format ID _____

Application control number _____ to _____

Batch ID _____

Date delivered to app . . _____ to _____

Time delivered to app . . _____ to _____

Depending on the option you selected from the Transaction Store Facility Menu (TF01), some of the fields will contain default values. Fields that do not apply to the option are not displayed.

Managing Your Data Using the Transaction Store Facility

To specify Application criteria, complete one or more of the following fields, and press Enter.

In this field:	Enter:
Application ID	An application ID defined in the activity log profile. For the DataInterchange Utility, the default ID is EDIFFS. For all other DataInterchange services, the default ID is EDIMP.
Data format ID	A data format ID.
Application control number	An application control number. The value must match exactly the application control value in the data, including upper and lower casing of characters.
Batch ID	A batch ID. The default batch ID is a date and time stamp in the form <i>ddhhmmss</i> .
Date delivered to app	A date or range of dates to select transactions by when they were translated to the application.
Time delivered to app	A time or range of times to select transactions by when they were translated to the application.

Store Status Criteria

TF02

Criteria Selections

The list of transactions you will work with depends on the selection

TF07

Store Status Criteria

Type the number of your choice and press Enter.

Choice ==> -

- Purge pending - date expired
- Purge pending - user requested
- Held
- Active

Select one of the following options:

Choice	Selects
1	Transactions the system has marked for purging because the store time expired
2	Transactions you or another user have marked for purging
3	Transactions on hold
4	Transactions neither marked for purging nor on hold

Additional Selection Criteria Panel (TF08)

The Additional Selection Criteria panel (TF08) is displayed after you enter your selection criteria. On this panel, you can do one of the following:

- To specify more selection criteria and have them added to the criteria you have already specified, type **Y**, and press Enter. Remember, DataInterchange retrieves only those transactions that meet **all** of the specified conditions.
- To select transactions from the Transaction Store using the criteria already entered, type **N**, and press Enter.
- To discard the criteria you have specified and enter new criteria, press F3 (Exit) or F12 (Cancel).

TF02
Criteria Selections

The list of transactions you will work with depends on the selection

TF08
Additional Selection Criteria

— Press Enter to begin your query, or answer Y to enter another set of selection criteria. Any additional criteria will be used to append transactions to those you have already selected. If you wish to modify your current criteria, press Exit or Cancel to return to the appropriate category of criteria.

— Do you wish to enter additional selection criteria? (Y/N) N

Sending and Receiving Transactions

The following options on the Transaction Store Facility Menu (TF01) provide services for enveloping, sending, receiving, and translating transactions.

- Envelope transactions
- Envelope and send transactions
- Receive and deenvelope transactions
- Translate received transactions
- Reenvelope transactions
- Reenvelope and send transactions
- Re-translate received transactions

Enveloping Transactions

The *Envelope transactions* option takes transactions from the Transaction Store, envelopes them, and puts the results in the transaction data queue specified in the network profile.

The *Reenvelope transactions* option does the same with transactions that were enveloped previously or were part of an unsuccessful enveloping request.

To envelope or reenvelope transactions, follow these steps:

1. From the Transaction Store Facility Menu (TF01), select *Envelope transactions* or *Reenvelope transactions*. The Criteria Selections panel (TF02) is displayed.

Managing Your Data Using the Transaction Store Facility

- Specify your selection criteria as described in “Selecting EDI Transactions You Want to Work With” on page 10-7, or press Enter twice to use the default criteria.

The Envelope Transactions panel (TF30) is displayed. :cp 10

Detail	Envelope	eVentlog	eXclude	Image	Line	Options	Print
TF30			Envelope Transactions				1 to 2 of 2
Asterisks (*) denote the controlling transaction in a related group.							
A	Ref	Trading Partner	Standard	Application	Transaction		
	Num	Nickname	Trans ID	Control Number	Status		
-	00001	PISCES	850	INVOICE2200000101011	TRANSLATED		
-	00002*	CAPRICORN	*****	INVOICE2200000211111	TRANSLATED		

- Select the transactions you want to envelope by typing **E** in the action column next to each transaction, and pressing Enter. You can also select transactions by typing **X** next to the transactions you *do not* want to envelope, typing **E** on the command line, and pressing Enter.

DataInterchange envelopes the transactions, putting them in the fewest possible functional groups and interchange envelopes. For details, see “Envelope” on page 10-32.

After DataInterchange envelopes the transactions, the Envelope Results panel (TF31) is displayed, listing the files created for this enveloping request.

TF31	Envelope Results		1 to 1 of 1
The following files were created as a result of your envelope request.			
Network	TP Nickname	File Name	
IINR3	PISCES	QDATA	

- Press F3 (Exit) to leave this screen. The Envelope Transactions panel (TF30) is redisplayed.
- Press F3 (Exit) to return to the Transaction Store Facility Menu (TF01).

Enveloping and Sending Transactions

The *Envelope and send transactions* option takes transactions from the Transaction Store, envelopes them, puts the results in the transaction data queue specified by the network profile member, then sends them to the network.

The *Reenvelope and send transactions* option does the same with transactions that were enveloped previously or were part of an unsuccessful enveloping request.

To envelope and send, or reenvelope and send transactions, follow these steps:

- From the Transaction Store Facility Menu (TF01), select *Envelope and send transactions* or *Reenvelope and send transactions*. The Criteria Selections panel (TF02) is displayed.
- Specify your selection criteria as described in “Selecting EDI Transactions You Want to Work With” on page 10-7, or press Enter twice to use the default criteria.

The Envelope and Send Transactions panel (TF40) is displayed.

Detail	eVentlog	eXclude	Image	Line	Options	Print	Send
TF40					Envelope and Send Transactions		1 to 2 of 2
Asterisks (*) denote the controlling transaction in a related group.							
A	Ref	Trading Partner	Standard	Application	Transaction		
	Num	Nickname	Trans ID	Control Number	Status		
-	00001	PISCES	850	INVOICE220101011111	TRANSLATED		
-	00002*	CAPRICORN	*****	INVOICE220101021111	TRANSLATED		

3. Select the transactions you want to envelope and send by typing **S** in the action column next to each transaction, and pressing Enter. You can also select transactions by typing **X** next to the transactions you *do not* want to envelope and send, typing **E** on the command line, and pressing Enter.

DataInterchange envelopes the transactions, putting them in the fewest possible functional groups and interchange envelopes. For details, see “Envelope” on page 10-32 for more information.

After DataInterchange envelopes the transactions, the Network Access panel (TF41) is displayed. This panel will be displayed for each envelope file created.

TF40	Envelope and Send Transactions	1 to 1 of 1
A	Ref	Transaction
-	Num	Status
-	001	TRANSLATED

TF41 Network Access

Network ID: IIN

Requestor ID _____

4. For each file, type the requestor ID in the *Requestor ID* field, and press Enter. The Envelope and Send Transactions panel (TF40) is redisplayed.
5. Press F3 (Exit) to return to the Transaction Store Facility Menu (TF01).

Receiving Transactions

The *Receive and deenvelope transactions* option receives transactions from the network indicated in a requestor profile member, then deenvelopes them, and puts them in the Transaction Store. If the transactions have already been received from the network and stored in a file, this option deenvelopes the transactions and puts them in the Transaction Store.

To receive transactions, follow these steps:

1. From the Transaction Store Facility Menu (TF01), select *Receive and deenvelope transactions*. The Receive Transactions panel (TF60) is displayed.

Managing Your Data Using the Transaction Store Facility

For DataInterchange for MVS, the panel looks like this:

TF60	Receive Transactions
To receive transactions from the network and place them in the Transaction Store, enter the requestor ID and a print file name.	
To place previously received transactions in the Store Facility, enter a receive file name and a print file name.	
Print file . . WORKERB.EDI.PRTFILE_____	
Requestor ID	_____
or	
Receive file	_____

The *Print file* field contains the default file name allocated to PRTFILE, indicating the file for any messages created by the deenveloper or translator while producing functional acknowledgments for received transactions. Optionally, these messages can be routed to an MQSeries Queue. Instead of specifying a fully-qualified name of a sequential file, enter the name of a DataInterchange MQSeries Queue profile member concatenated with :MQ. For example, if you have a DataInterchange MQSeries Queue profile member named *MQPRINT*, you would enter:

MQPRINT:MQ

DataInterchange will then route the messages to the associated MQSeries Queue.

For DataInterchange for CICS, the panel looks like this:

TF60	Receive Transactions
To receive transactions from the network and place them in the Store Facility, enter the requestor ID and a print file name and type.	
To place previously received transactions in the Store Facility, enter a storage file name and a print file name and type.	
Print file name	_____
Print file type	_____
Requestor ID	_____
or	
Receive data name	_____

The *Print file name* field contains the name of a temporary storage queue, transient data queue, the ddname of a VSAM entry sequenced data set, or the name of a DataInterchange MQSeries Queue profile member for messages.

The *Print file type* field identifies the type of print file: TS for temporary storage queue (auxiliary storage), TM for temporary storage queue (main storage), TD for transient data queue, VS for VSAM entry sequenced data set, or MQ for DataInterchange MQSeries Queue profile member.

2. To receive transactions from a network mailbox, in the *Requestor ID* field, type the name of the requestor profile member that identifies the mailbox and provides other information about the transactions you want to receive, and press Enter.

3. To deenvelope transactions previously received from the network and stored in a file:

- For DataInterchange for MVS, type the name of the file in the *Receive file name* field, and press Enter.

- For DataInterchange for CICS, type the name of a temporary storage queue in the *Receive data name* field, and press Enter.

If you are receiving transactions from the network, and the network allows you to retrieve selected transactions, the Network Receive Options panel (TF61) is displayed.

TF60		Receive Transactions	
To receive transactions from the network and place them in the Store Facility, enter the requestor ID and a print file name.			
To	TF61	Network Receive Options	enter
a			
r			
Pri	To receive all transactions from your mailbox, leave the options blank and press Enter.		_____
Req	From trading partner nickname _____		
Rec	Specific name _____		_____

4. Do one of the following to receive transactions:

- To receive all transactions in the mailbox, leave both fields blank, and press Enter.
- To receive all transactions from a trading partner, type the trading partner nickname in the *From trading partner nickname* field, and press Enter.
- To receive specific transactions identified by trading partner agreement, type the name, or message user class, that identifies the specific transactions in the *Specific name* field, and press Enter.

The Receive Transactions panel (TF60) is redisplayed.

5. Press F3 (Exit) to return to the Transaction Store Facility Menu (TF01).

Translating Received Transactions

The *Translate received transactions* option takes transactions from the Transaction Store, translates them to application format, then delivers them to a specified destination.

The *Re-translate received transactions* option provides the same services for transactions that were previously translated.

To translate or re-translate received transactions, follow these steps:

1. From the Transaction Store Facility Menu (TF01), select *Translate transactions* or *Re-translate transactions*. The Criteria Selections panel (TF02) is displayed.
2. Specify your selection criteria as described in “Selecting EDI Transactions You Want to Work With” on page 10-7, or press Enter twice to use the default criteria.

The Translate Received Transactions panel (TF50) is displayed.

Managing Your Data Using the Transaction Store Facility

3. Select the transactions you want to translate by typing **T** in the action column next to each transaction, and pressing Enter.

Detail	eVentlog	eXclude	Image	Line	Options	Print	Translate
TF50		Translate Received Transactions					1 to 2 of 2
A	Ref	Trading Partner	Standard	Interchange	Transaction		
	Num	Nickname	Trans ID	Control Number	Control Number		
t	1	PISCES	850	00000011000023	00000022222220		
-	2	CAPRICORN	850	00000011000024	00000022222221		

You can also select transactions by typing **X** next to the transactions you *do not* want to translate, typing **T** on the command line, and pressing Enter.

The Data Destination panel (TF51) is displayed.

4. To select the data destination:
 - a. Type **Y** and press Enter to send the data to the Interactive Entry Facility database, from which you can view or print the data.
 - b. Type **N** and press Enter to send the data to the application file specified by the transaction's application data format or trading partner usage.

TF50		Translate Received Transactions				1 to 1 of 1
A	Ref	Trading Partner	Standard	Interchange	Transaction	
-						

TF51 Data Destination

To store the translated data as a business document in the Interactive Entry Facility, type Y and press Enter. To store it in the application file, type N and press Enter.

Store as IEF document? (Y/N) **n**

If you decided to store the data in an application file, the Specify Exception and Print Files panel (TF52) is displayed.

The exception file holds translated transactions that cannot be stored in the application file, for example, if the system is unable to open the application file. The exception file's definition must take into account the largest record you expect to receive. The default exception file for MVS is the file allocated to ddname FFSEXCP.

The print file holds the report summarizing the results of the translation. The default print file for MVS is the file allocated to the ddname PRTFILE.

For DataInterchange for MVS, the panel looks like this:

TF50		Translate Received Transactions			1 to 1 of 1
A	Ref	Trading Partner	Standard	Interchange	Transaction
-	TF52	Specify Exception and Print Files			
Type the file names for the exception and print files and press Enter.					
Exception		_____			
Print . .		_____			

5. For DataInterchange for MVS, do the following:

- Enter the exception file name in the *Exception* field.
- Enter the print file name in the *Print* field.

Note: Both the exception file and print file can be MQSeries Queues. Instead of specifying a fully qualified name of a sequential file, enter the name of a DataInterchange MQSeries Queue profile member concatenated with :MQ. For example, if you have a DataInterchange MQSeries Queue profile member named *MQPRINT*, you would enter:

MQPRINT:MQ

DataInterchange will then route the messages to the associated MQSeries Queue.

- Press Enter.

For DataInterchange for CICS, the panel looks like this:

TF50		Translate Received Transactions			1 to 1 of 1
A	Ref	Trading Partner	Standard	Interchange	Transaction
-	TF52	Specify Exception and Print Files			
Type the file names for the exception and print files and press Enter.					
Exception name		_____			
Exception type		_____			
Print name . .		_____			
Print type . .		_____			

6. For DataInterchange for CICS, do the following:

- Enter the exception file name in the *Exception file name* field.
- Enter the exception file type (TS, TM, TD, VS, or MQ) in the *Exception file type* field.

Managing Your Data Using the Transaction Store Facility

- c. Enter the print file name in the *Print file name* field.
- d. Enter the print file type (TS, TM, TD, VS, or MQ) in the *Print file type* field.
- e. Press Enter.

The Additional Record Options panel (TF53) is displayed.

7. Type a slash (/) next to each information record you want delivered with the transaction data, and press Enter.

TF50

Translate Received Transactions

1 to 1 of 1

T

TF53

Record format and Additional Record Options

Use a slash (/) to indicate that the application data should be written in RAWDATA format versus 'C' and 'D' records

Rawdata format

To receive these records along with your application data, type a slash (/) next to the records you want and press Enter.

Information record

Envelope header record

Group header record

Transaction header record

Queuing totals

The requested records follow the transactions with which they are associated. The first character in the record is the record type code.

Code	Record	Contains
I	Information record	Interchange, group, and transaction envelope segments
E	Envelope header record	Interchange header segment
G	Group header record	Group header segment
T	Transaction header record	Transaction set or message header segment
Q	Queuing totals	Totals for number of bytes, segments, transaction sets or messages, and groups in the interchange

For more information about these records, see *DataInterchange Programmer's Reference*.

The Translate Received Transactions panel (TF50) is redisplayed.

8. Press F3 (Exit) to return to the Transaction Store Facility Menu (TF01).

Managing Transactions

The following options on the Transaction Store Facility Menu (TF01) provide services to track transactions that pass through the Transaction Store.

- Report transaction status
- Update store status
- Report interchange/group status

- Update interchange/group status
- Update network status

The *Update network status* option lets you retrieve network acknowledgments and update Transaction Store records accordingly.

Reporting Transaction Status

The *Report transaction status* option lets you obtain detailed or summary information about transactions on screen, in a file, or on paper. On a printed report, the cover page lists the criteria you entered to select items appearing in the report.

To report transaction status, follow these steps:

1. From the Transaction Store Facility Menu (TF01), select *Report transaction status*. The Criteria Selections panel (TF02) is displayed.
2. Specify your selection criteria as described in “Selecting EDI Transactions You Want to Work With” on page 10-7, or press Enter twice to use the default criteria.

The Report Transaction Status panel (TF10) is displayed.

3. Type **R** in the action column next to any item, and press Enter.

Acknowledgment	Detail	eVentlog	Image	Line	Options	Print	Report	Summary
TF10		Report Transaction Status					1 to 4 of 4	
Asterisks (*) denote the controlling transaction in a related group.								
A	Ref	Trading Partner	Standard	Transaction	Transaction			
	Num	Nickname	Trans ID	Control Number	Status			
r	1	PISCES	810	00000000000111	DELIVERED			
-	2	PISCES	850	00000000000222	RECEIVED			
-	3*	CAPRICORN	*****	*****	TRANSLATED			
-	4	PISCES	810	00000000000333	SENT			

The Report Selections panel (TF14) is displayed.

4. Type a slash (/) next to each report you want to print, and press Enter.

TF10

Report Transaction Status

1 to 4 of 4

Asterisks (*) denote the controlling transaction in a related group.

A	Ref	Trading Partner	Standard	Transaction	Transaction
	Num	Nickname	Trans ID	Control Number	Status

r

-

-

TF14

Report Selections

Place a slash (/) next to the reports you want to print and press Enter.

Transaction activity summary report

☒ Transaction status summary report

Managing Your Data Using the Transaction Store Facility

The Transaction activity summary report shows the various totals for all the transactions that match the selection criteria. You can view the same information using the *Summary* action on the Report Transaction Status (TF10) panel.

The Transaction status summary report shows the transaction, network, and store status for all the transactions that match the selection criteria. An example of the printed report is shown in Figure 10-3.

After printing the requested reports, the Report Transaction Status panel (TF10) is redisplayed.

5. Press F3 (Exit) to return to the Transaction Store Facility Menu (TF01).

TF80	Status Summary Report for Outbound Transactions					Date: 97/12/31 Time: 08:12:53
Transaction Handle Date Enveloped	Trading Partner Nickname Interchange Cntrl No	Data Format ID Network Status	Transaction Status Group Control No	Store Status Func Ack Status		
19981110093012000001 97/12/26	PISCES 00000000001212	PISCESPOSEND Accepted by network	Transaction accepted 00000000001321	Purge Requested Received		
19981212104533000001 97/12/22 97/12/12	PISCES 00000000001013 00000000000533	PISCESINVSEND Not sent - net error Recall request error	Send translate error 00000000001934 00000000001073	Active Received		
19981110103311000001 97/12/20	CAPRICORN 00000000000852	CAPRICORNPOSEND Accepted by network	Transaction accepted 00000000001321	Active Received		
19981110103311000001 97/12/20	CAPRICORN 00000000000843	CAPRICORNINVSEND Accepted by network	Transaction accepted 00000000001934	Active Received		
TF80	Status Summary Report for Inbound Transactions					Date: 97/11/26 Time: 12:12:12
Transaction Handle Date Translated	Trading Partner Nickname Data Format ID	Standard Trans ID Translation Status	Transaction Status	Store Status		
19981226134427000001 R 97/12/26	PISCES PISCESPORECV	850 Acceptable	Receive translated	Active		
19981222103259000001 97/12/26 97/12/22	CAPRICORN CAPRICORNPORECV CAPRICORNPORECV2	850 Acceptable Unacceptable	Receive translated	Active		

Figure 10-3. Status Summary Report. An R after the transaction handle indicates the transaction is a related transaction and is part of a bundle.

Updating Store Status

The *Update store status* option lets you do the following:

- Mark transactions for purging
- Change transactions from purge status
- Place transactions on hold so that they are not affected by subsequent requests
- Release transactions from hold

To update store status, follow these steps:

1. From the Transaction Store Facility Menu (TF01), select *Update store status*. The Criteria Selections panel (TF02) is displayed.
2. Specify your selection criteria as described in “Selecting EDI Transactions You Want to Work With” on page 10-7, or press Enter twice to use the default criteria.

The Update Store Status panel (TF20) is displayed.

Detail	eVentlog	eXclude	Hold	Image	Line	Options	Print	purGe	Release
Unpurge									
TF20			Update Store Status					1 to 2 of 2	
Asterisks (*) denote the controlling transaction in a related group.									
A	Ref	Trading Partner	Standard	Transaction			Store		
	Num	Nickname	Trans ID	Status			Status		
—	1	PISCES	880	SEND REQUESTED			ACTIVE		
—	2*	CAPRICORN	850	SEND TRANSLATE ERROR			HELD		

- To change the store status of transactions, indicate the action in the action column, and press Enter. You can also change store status by typing **X** next to the transactions you *do not* want to change, indicating the action on the command line, and pressing Enter.

Enter: To:

- H Place transactions on hold
- R Release transactions from hold
- G Mark transactions for purging
- U Restore transactions marked for purging to their previous status

For a list of valid transaction store services status codes, see Table 10-4 on page 10-29.

- When you are through updating the store status of the selected transactions, press F3 (Exit) to return to the Transaction Store Facility Menu (TF01).

Updating Network Status

When you select the *Update network status* option from the Transaction Store Facility Menu (TF01), the system:

- Retrieves network acknowledgments from each network defined in the network profile.
- Matches received acknowledgments with transactions pending acknowledgment.
- Updates the status information accordingly.

Because this work occurs in the background, expect some delay before the status information you see is updated.

Reporting Interchange or Group Status

To report interchange or group status, follow these steps:

- From the Transaction Store Facility Menu (TF01), select *Report interchange/group status*. The Criteria Selections panel (TF02) is displayed.
- Specify your selection criteria as described in “Selecting EDI Transactions You Want to Work With” on page 10-7, or press Enter twice to use the default criteria.

The Report Interchange Status panel (TF80) is displayed.

Managing Your Data Using the Transaction Store Facility

3. Type **R** in the action column next to any item, and press Enter.

Detail	Image	Line	Print	Report	Summary	gr0ups	Transactions
TF80		Report Interchange Status					1 to 2 of 2
A	Ref	Trading Partner	Interchange		Network	Ack	
	Num	Nickname	Control Number		Status	Pnd	
r	1	PISCES	00000011000023		SEND	ENVELOPED	
-	2	CAPRICORN	00000011000024		RECEIVE	RECEIVED	
						N	

The Report Selections panel (TF14) is displayed.

4. Type a slash (/) next to each report you want to print, and press Enter.

TF80		Report Interchange Status					1 to 2 of 2
A	Ref	Trading Partner	Interchange		Network	Ack	
	Num	Nickname	Control Number		Status	Pnd	
r						N	
-						N	

TF14 Report Selections

Place a slash (/) next to the reports you want to print and press Enter.

☒ Transaction activity summary report

☒ Transaction status summary report

After printing the requested reports, the Report Interchange Status panel (TF80) is redisplayed.

5. To report functional group status for an interchange, type **O** in the action column next to the interchange, and press Enter.

The Report Group Status panel (TF85) is displayed.

6. Type **R** in the action column next to any item, and press Enter.

Detail	Image	Line	Print	Report	Summary	Transactions
TF85				Report Group Status		1 to 2 of 2
A	Ref	Trading Partner	Interchange		Group	Functional Ack
	Num	Nickname	Control Number		Control Number	Status
<div>r</div>	1	PISCES	00000011000023		10000011000023	PENDING
	2	CAPRICORN	00000011000024		10000011000024	RECEIVED

The Report Selections panel (TF14) is displayed.

7. Type a slash (/) next to each report you want to print. The Report Group Status panel (TF85) is displayed.
8. Press F3 (Exit) twice to return to the Transaction Store Facility Menu (TF01).

Updating Interchange or Group Status

To update interchange or group status, follow these steps:

1. From the Transaction Store Facility Menu (TF01), select *Update interchange/group status*. The Criteria Selections panel (TF02) is displayed.
2. Specify your selection criteria as described in “Selecting EDI Transactions You Want to Work With” on page 10-7, or press Enter twice to use the default criteria. The Update Interchange Status panel (TF81) is displayed.

Detail	eXclude	Hold	Image	Line	Print	purGe	Release	Unpurge	grOups
Transactions		reConstruct							
TF81		Update Interchange Status						1 to 2 of 2	
A	Ref	Trading Partner	Interchange	Control Number	Direction	Network	Status	Ack	Pnd
Num	Nickname								
—	1	PISCES	00000011000023	SEND	TRANSLATED			N	
—	2	CAPRICORN	00000011000024	RECEIVE	RECEIVED			N	

3. To change the status of all transactions in an interchange, indicate the action in the action column, and press Enter. You can also change store status by typing **X** next to the interchange you *do not* want to change, indicating the action on the command line, and pressing Enter.

Enter: To:

- H Place all transactions in the interchange on hold
- R Release transactions in the interchange from hold status
- G Mark all transactions in the interchange for purging
- U Restore transactions in the interchange marked for purging to their previous status
- C Reconstruct or rebuild an interchange and write the interchange to the file associated with the network (QDATA). Reconstruct only works on a single interchange at a time.

For a list of valid transaction store services status codes, see Table 10-4 on page 10-29.

4. To update the status of transactions in the functional groups of the interchange, type the letter **O** in the action column next to the interchange, and press Enter. The Update Group Status panel (TF86) is displayed.

Detail	eXclude	Hold	Image	Line	Print	purGe	Release	Unpurge	
Transactions									
TF86		Update Group Status						1 to 2 of 2	
A	Ref	Trading Partner	Interchange	Group	Functional	Ack			
Num	Nickname		Control Number	Control Number	Status				
—	1	PISCES	00000011000023	10000011000023	PENDING				
—	2	CAPRICORN	00000011000024	10000011000024	RECEIVED				

5. To change the status of all transactions in a functional group, indicate the action in the action column, and press Enter. You can also change status by typing **X** next to the functional group you *do not* want to change, indicating the action on the command line, and pressing Enter.

Managing Your Data Using the Transaction Store Facility

Enter: To:

- H Place all transactions in the functional group on hold
- R Release transactions in the functional group from hold status
- G Mark all transactions in the functional group for purging
- U Restore transactions in the functional group marked for purging to their previous status

For a list of valid transaction store services status codes, see Table 10-4 on page 10-29.

6. Press F3 (Exit) to return to the Transaction Store Facility Menu (TF01).

Transaction Status Codes

Table 10-3 describes the transaction status codes. On the criteria panels, you specify transaction status by a two-digit code. On the list panels, such as Envelope Transactions, DataInterchange displays the transaction status name.

Table 10-3 (Page 1 of 2). Transaction Status Codes

Status Code	Name	Description
Send Transactions		
20	Send translate error	The translation error level is higher than the error level specified in the trading partner usage for the transaction mapping.
21	Send translated	The document was translated successfully. If translation errors occurred, the error level is less than or equal to the error level specified in the trading partner usage for the transaction mapping.
29	Trx detached - send	The envelope for this transaction was replaced by another envelope with an identical key. This transaction is not included in the new envelope. In effect, the transaction is detached from its envelope. This can occur if you reset the control numbers in the trading partner profile. If you still want to use this transaction, you can re-envelope or re-envelope and send it.
30	Enveloped	The transaction is enveloped and ready for sending.
31	Envelope error	An error occurred during enveloping. The document remained in the Transaction Store.
41	Sent with errors	The network processed the send command with minor errors.
42	Send request error	DataInterchange encountered an error while attempting to issue a send command to the network.
43	Not sent -- network error	The network encountered errors that prevented processing of the send command.
46	Send started	The network has started processing the send command.
48	Send requested	DataInterchange successfully issued a send command to the network.
49	Sent to network	DataInterchange successfully sent a message to the network.
50	Accepted by network	DataInterchange received a network receipt acknowledgment.
51	Delivered by network	DataInterchange received a network delivery acknowledgment.
52	Purged by network	DataInterchange received a purge acknowledgment from the network.
53	Recall requested	DataInterchange successfully issued a recall command.

Table 10-3 (Page 2 of 2). Transaction Status Codes

Status Code	Name	Description
54	Recall request error	DataInterchange encountered an error while attempting to issue a recall command to the network.
55	Recalled	DataInterchange received a cancel acknowledgment from the network.
61	Transaction accepted	DataInterchange received a functional acknowledgment indicating that the document is acceptable to the receiver.
62	Transaction rejected	DataInterchange received a functional acknowledgment indicating that the document is not acceptable to the receiver.
63	Transaction accepted with errors	DataInterchange received a functional acknowledgment indicating that the document was acceptable, but that errors were noted.
Receive Transactions		
70	Received	The document was successfully received from the network and deenveloped.
71	Received syntax error	A syntax error was found in the transaction during deenvelope. This is usually a control number mismatch between the transaction header and trailer, or an invalid segment count in the trailer.
72	Receive translated	The document was translated successfully. If translation errors occurred, the error level is equal to or less than the error level specified in the trading partner usage for the transaction mapping.
73	Receive translate error	The translation error level is higher than the error level specified in the trading partner usage for the transaction mapping. The standard data remains in the Transaction Store.
74	Trx detached - recv	An envelope was received that has the same key as the envelope that contained this transaction. This transaction is not included in the new envelope. In effect, the transaction is detached from its envelope. This can occur when your trading partner sends the same envelope more than once. You cannot translate or re-translate this transaction.

A transaction's status determines which services are valid. Table 10-4 shows which Transaction Store Facility services are valid for each status code, where X indicates that the service is valid. For example, if the transaction's status is enveloped (30), you can reenvelope or re-send the transaction, but you cannot send it.

Table 10-4 (Page 1 of 2). Valid Store Services for Status Codes

Status Code	Tran	Re-T	Env	Re-E	Send	Re-S	Hold	Purg	Unp	Rel
Send trans error (20)							X	X		
Send translated (21)			X		X		X	X		
Trx detached - send (29)				X		X	X	X		
Enveloped (30)				X		X	X	X		
Envelope error (31)				X		X	X	X		
Sent with errors (41)				X		X	X	X		
Send request error (42)				X		X	X	X		
Not sent net error (43)				X		X	X	X		
Send requested (48)				X		X	X	X		
Sent to network (49)				X		X	X	X		

Managing Your Data Using the Transaction Store Facility

Table 10-4 (Page 2 of 2). Valid Store Services for Status Codes

Status Code	Tran	Re-T	Env	Re-E	Send	Re-S	Hold	Purg	Unp	Rel
Accepted by network (50)				X		X	X	X		
Delivered by network (51)				X		X	X	X		
Purged by network (52)				X		X	X	X		
Recall requested (53)				X		X	X	X		
Recall request error (54)				X		X	X	X		
Recalled (55)				X		X	X	X		
Transaction accepted (61)				X		X	X	X		
Transaction rejected (62)				X		X	X	X		
Transaction accepted with errors (63)				X		X	X	X		
Received (70)	X						X	X		
Received syntax error(71)							X	X		
Receive translated (72)		X					X	X		
Receive trans error (73)		X					X	X		
Trx detached - recv (74)							X	X		
Held (1)										X
Purge - date expired (2)							X	X		
Purge - user request (4)							X		X	

Tran = Translate
 Re-T = Re-translate
 Env = Envelope
 Re-E = Reenvelope
 Send = Envelope and send

Re-S = Reenvelope and send
 Hold = Place transactions on hold
 Rel = Release transactions from hold
 Purg = Mark for transactions for purging
 Unp = Release transactions from purge

Panel Actions

This section describes the actions you can perform from the action bar of the Transaction Store Facility panels. Not all actions are available on all panels.

Acknowledgment

This action displays an image of the functional acknowledgment for a transaction or a group of related transactions. The image also shows the trading partner name, transaction handle, and transaction status.

TF13	Functional Acknowledgment Image	1 to 1 of 1
Trading partner nickname : PISCES		
Transaction handle . . . : 19981214154623000023		
Transaction status . . . : TRANSACTION ACCEPTED		
AK1*IN*40088!AK2*810*000048118!AK5*A!AK9*A*1*1*1!		

Detail

This action displays detailed information about a transaction, functional group, or interchange.

For example, the Transaction Detail panel (TF73) is displayed when you use *Detail* from the Report Transaction Status panel (TF10).

TF73	Transaction Detail	Page 1 of 2
Transaction handle : 19981214151318000101		
Trading partner nickname . . . : PISCES		
Internal trading partner ID : Pisces Fish Company		
Direction : SEND		
Data format ID : PISCESINVSND		
Application control number . . : P0142536		
Interchange control number . . : 00000000000022		
Group control number : 00000000000027		
Transaction control number . . : 00000000027251		
Transaction status : TRANS REJECTED		
Added to store : 97/12/14-14:34:20		
Store status : ACTIVE		
Delivered to application . . . :		
Translation error level . . . : 1		
Translation : ACCEPTABLE		
Network ack requested . . . : D		
Network status : DELIVERED		
Functional acknowledgment . . : RECEIVED		

Use the *Options* action to select the details you want to see: transaction, functional group, and/or interchange.

Related Transactions: An asterisk after the number in the *Ref Num* column indicates a bundle of transactions. A *bundle* is a set of transactions treated as a unit. A list panel, such as Envelope and Send Transactions (TF40), displays only the first, or *controlling*, transaction in the bundle. For more information about bundles, see the *DataInterchange Programmer's Reference*.

If you use *Detail* for a bundle of transactions, the Related Transactions panel (TF71) is displayed.

Managing Your Data Using the Transaction Store Facility

Detail	eVentlog	Image	Print
<hr/>			
TF40	Envelope and Send Transactions		1 to 1 of 1
Asterisks (*) denote the controlling transaction in a related group.			
A Re	TF71	Related Transactions	1 to 4 of 4
Nu			
d	A	Transaction Handle	Transaction Control Number
			Standard Trans ID
	-	19981214153127000000	00000000000023 850
	-	19981214153127000001	00000000000024 850
	-	19981214153127000002	00000000000026 850
	-	19981214153127000003	00000000000027 850

Envelope

This action creates the envelope segments for the transactions you are sending, and puts the transactions in a file ready to be sent to the network. The enveloper sorts the transactions to create the fewest number of functional groups and interchange envelopes. When the value of the following items change, DataInterchange starts a new functional group or interchange envelope:

New interchange

Network ID
Trading partner nickname
Envelope type
Interchange version/release
Interchange receiver ID
Interchange sender ID
Interchange password
Interchange application reference
Internal trading partner ID
Any delimiter

New functional group

Functional group ID
Group version
Group release
Group security profile member name
Group encryption key name
Group authentication key name
Application sender ID
Application receiver ID
Group password

The Internal trading partner ID starts a new interchange only when you are using the DataInterchange Utility or an application program interface. To envelope by internal trading partner ID using the Transaction Store Facility, select only the transactions that have the same internal trading partner ID.

A new interchange will be started when there is a switch from application generated message control numbers to DataInterchange generated message control numbers.

Note: One way to signal the start of a new functional group is to change the application sender ID, the application receiver ID, or the envelope profile member name in the trading partner send usage. A change in any of these fields will start a new group, although they are not included in the sort key used by the enveloper. When using the utilities, the enveloper gets the application sender ID and application receiver ID from the override values in the C record. When using the application program interface, the enveloper gets the application sender ID and application receiver ID from the translator interface control block.

If all the transactions you selected are enveloped in the same functional group or interchange, they are in the order in which they were entered into the Transaction Store.

You can request functional group envelopes without the interchange envelopes by providing a blank sender ID in the envelope data.

If you use *Envelope* on the command line with no reference number, all listed transactions are enveloped except those that have been marked with *eXclude*.

The Envelope Results panel (TF31) lists the envelope files that are created as a result of your request.

For more information, see “Enveloping Transactions” on page 10-15.

eVentlog

This action displays the event log entries that are associated with this transaction. From the Transaction Store Facility, you can view only those log entries that are associated with a transaction, such as transaction level errors and transaction level application data. You cannot view log entries from any other level. Enveloping errors and standard image entries are at the envelope level, and cannot be viewed from the Transaction Store Facility.

For example, when you use the *eVentlog* action, the transaction handle is used as the associated entry ID to retrieve and display the event log entries on the Event Log Entries panel (EL03). If you use *eVentlog* for a bundled transaction, the Related Transactions panel (TF71) is displayed.

Line Sort toggle View				
<hr/>				
EL03		Event Log Entries		1 to 5 of 5
Application ID: EDIMP				
A	Line	Date	Time	Log data
—	0001	97/12/07	10:23:44	PS0020*** This is an error log record
—	0002	97/12/07	10:34:12	TR0001*** This is an error log for Translator
—	0003	97/12/07	10:37:33	Transaction data queued
—	0004	97/12/07	10:37:43	Data send Network
—	0005	97/12/07	10:38:02	Ack delivered

eXclude

This action excludes an item from an action entered on the command line. If you want to affect all but a few of the listed items, use *eXclude* to mark those you *do not* want to affect, then enter the action on the command line with no reference number.

This action can be used only in the action column with one of the following actions on the command line: *Envelope*, *Send*, *Translate*, *Hold*, *purGe*, *Release*, or *Unpurge*.

grOups

This action displays a list of the functional groups in an interchange.

Hold

This action places an item on hold. While on hold, an item’s status cannot be changed, nor is the item purged when the store time expires.

If you use *Hold* on the command line with no reference number, all listed items are placed on hold except those that have been marked with *eXclude*.

All items can be placed on hold except those already on hold. If you try to hold items already being held, a message is displayed and the status does not change.

Image

This action displays an image of an interchange's header and trailer segments, a functional group's header and trailer segments, or a transaction without the envelope segments.

For example, when you use the *Image* on the Report Transaction Status panel, the Transaction Image panel (TF75) is displayed. If you use *Image* on a bundled transaction, the Related Transactions panel (TF71) is displayed for you to select an individual transaction.

You can find a text string within the image using the FIND command, which will display the row where the text occurs.

```
Find
-----
TF75                      Transaction Image                      1 to 12 of 18

Trading partner           : TPTSU16A
Transaction handle      . . : 19991231212304450000
Transaction status      . . : ENVELOPE ERROR

BEG*00*ST*P00000100001**920331*REF000001001!
CSH**B*10285*ACCT10001009!
ITA*C***MH**6.375*299!
N1*BT*J D PIERPONT!
N3*123 BUSCH BLVD W!
N4*TAMPA*FL*34707!
N1*ST*LUCY AND DON HO!
N3*3554 W PILLSBURY ROAD!
N4*PENSACOLA*FL*32664!

N1*AG*THE BARBER OF SEVILLE!
N3*249 E HOLLYWOOD BLVD!
N4*HOLLYWOOD*CA*21125!
```

Line

This action scrolls to a specific line, using the value in *Ref Num* column. For example, to make line 500 the first line on the panel, type **L 500** on the command line, and press Enter.

Options

This action displays the Detail and Image Options panel (TF89) to allow you to select the details you want to see (transaction, functional group, and/or interchange) when you use the *Detail* action.

TF10		Report Transaction Status	1 to 3 of 3
Asterisks (*)	<div> <div>TF89</div> <div>Detail and Image Options</div> <div>Place a slash (/) next to the type of details that you want</div> <div> <input checked="" type="checkbox"/> Transaction details <input type="checkbox"/> Group details <input type="checkbox"/> Interchange details </div> <div>Indicate with a slash (/) if you want to display one segment per line for the image only or image merged with func ack image:</div> <div> <input checked="" type="checkbox"/> Segmented image <input type="checkbox"/> Segmented image merged with func ack image (997 or CONTROL) </div> </div>		

Print

This action prints one or more of the following to the print file specified when DataInterchange was started:

- Transaction image
- Transaction details
- Event log entries
- Functional group detail and image
- Interchange detail and image

For example, when you use the *Print* action on the Report Transaction Status panel (TF10), the Print Selections panel (TF72) is displayed.

TF10		Report Transaction Status	1 to 6 of 6
Asterisks (*) denote the controlling transaction in a related group.			
A	Ref	Tra	<div> <div>TF72</div> <div>Print Selections</div> <div>Reference Number: 1</div> <div>Trading Partner : PISCES</div> <div>Place a slash (/) next to the items you want to print and press Enter.</div> <div> <input type="checkbox"/> Transaction image <input type="checkbox"/> Transaction details <input type="checkbox"/> Event log images <input type="checkbox"/> Functional acknowledgment image </div> </div>
	Num	Nic	
p	1	PIS	
-	2	PIS	
-	3	CIS	
-	4	PIS	
-	5	CAP	
-	6	CAP	

If you use *Print* on a bundled transaction, the Related Transactions panel (TF71) is displayed for you to select an individual transaction.

Images will be printed as either segmented images, merged images, or wrapped images based on what was last specified in the Detail and Image Options panel (TF89).

purGe

This action marks an item for purging from the Transaction Store.

If you use *purGe* on the command line with no reference number, all listed items are placed in user-purged status except those that have been marked with *eXclude* or are on hold.

All items can be marked for purging except those that are already in user-purged status or on hold. If you try to purge items that are not eligible, a message is displayed and no change in status occurs.

The system automatically marks transactions for purging when they have been in the Transaction Store for 30 days, so that the store status becomes *Purge (store time expired)*. You can override the system default using the PURGINT keyword in the DataInterchange Utility. See the *DataInterchange Programmer's Reference* for details.

Release

This action restores a held item to its prior status or to store-time-expired status if the store time expired during the hold period.

If you enter *Release* on the command line with no reference number, all held items are released except those marked with *eXclude*.

If you try to release items that are not eligible, a message is displayed and the status does not change.

Report

This action displays the Report Selections panel (TF14), from which you can select reports for printing. This panel is described on page 10-23.

Send

This action envelopes and sends transactions. The enveloper sorts the transactions to create the fewest functional groups and interchange envelopes possible. For more information about how the enveloper creates functional groups and interchanges, see "Envelope" on page 10-32.

You can request functional group envelopes without the interchange envelopes by providing a blank sender ID in the envelope data.

If you use *Send* on the command line with no reference number, all listed transactions are enveloped and sent except those that have been marked with *eXclude*.

Summary

This action displays the Transaction Activity Summary panel (TF11), which is a summary of activity for inbound and outbound transactions in an interchange, functional group, or transaction list.

TF11 Transaction Activity Summary				
Outbound Status		Count	Inbound Status	Count
Selected transactions		1940	Selected transactions . . .	1000
Translation		1940	Acceptably translated . .	797
Acceptably translated . .		1927	Unacceptably translated:	13
Unacceptably translated .		13	Not yet translated . . .	190
Enveloping		1000	Detached	0
Enveloped		923		
Enveloping errors		77		
Send requests		923		
Sent		800		
Pending functional ack . .		10		
Pending network ack . . .		10		
Not sent		123		
Error on request to send:		106		
Network error		27		
Detached		0		

Transactions

This action displays a list of transactions within an interchange or functional group.

For example, when you use *Transactions* on the Report Interchange Status panel, the Report Transaction Status panel (TF10) is displayed.

Translate

This action translates a received transaction to application format, and delivers the translated data either to the IEF or to an application file, whichever you indicate on the Data Destination panel (TF51).

If you enter *Translate* on the command line with no reference number, all listed transactions are translated except those that have been marked with *eXclude*.

Unpurge

This action restores an item from user-purged status to its prior status, or to store-time-expired status if the store time expired and no acknowledgments are pending. Until the purge batch job runs, you can restore the items. After the purge job runs, you cannot restore them.

If you use *Unpurge* on the command line with no reference number, all user-purged items are restored except those that have been marked with *eXclude*.

If you try to unpurge items that are not eligible, a message is displayed and no change in status occurs.

Chapter 11. Exporting and Importing Transactions

Overview of the Export and Import Process	11-1
Export and Import File List	11-3
Export and Import Restrictions and Limitations	11-3
Using the Export Menu (EI00) Options	11-4
Exporting Standards	11-5
Exporting Application Data Formats	11-7
Exporting Trading Partner Transactions (Maps)	11-7
Exporting Control Strings	11-10
Exporting Business Document Definitions	11-12
Exporting Business Document Layouts	11-14
Exporting Profiles	11-14
Exporting Tables	11-16
Using the Import Menu (EI01) Options	11-16
Importing Standards	11-17
Importing Application Data Formats	11-18
Importing Trading Partner Transactions (Maps)	11-19
Importing Control Strings	11-20
Importing Business Document Definitions	11-21
Importing Business Document Layouts	11-22
Importing Profiles	11-22
Importing Tables	11-23
Printed Report	11-24
Export/Import Facility and Utility Comparisons	11-25

Chapter 11. Exporting and Importing Transactions

Export and import can save time and effort by allowing you and your trading partners to exchange DataInterchange setup files. For example, a manufacturer can export the mapping of a purchase order to a supplier so that the supplier does not have to recreate the purchase order information. Export and import also provide an effective way of promoting your work from your test system to your production system.

Overview of the Export and Import Process

The export function extracts the setup data you select. You can then send the extracted data through the network to your trading partner's system. The trading partner then receives and imports (inserts) the data into their database. DataInterchange provides reports to record exported and imported data. Figure 11-1 on page 11-2 illustrates the export and import process.

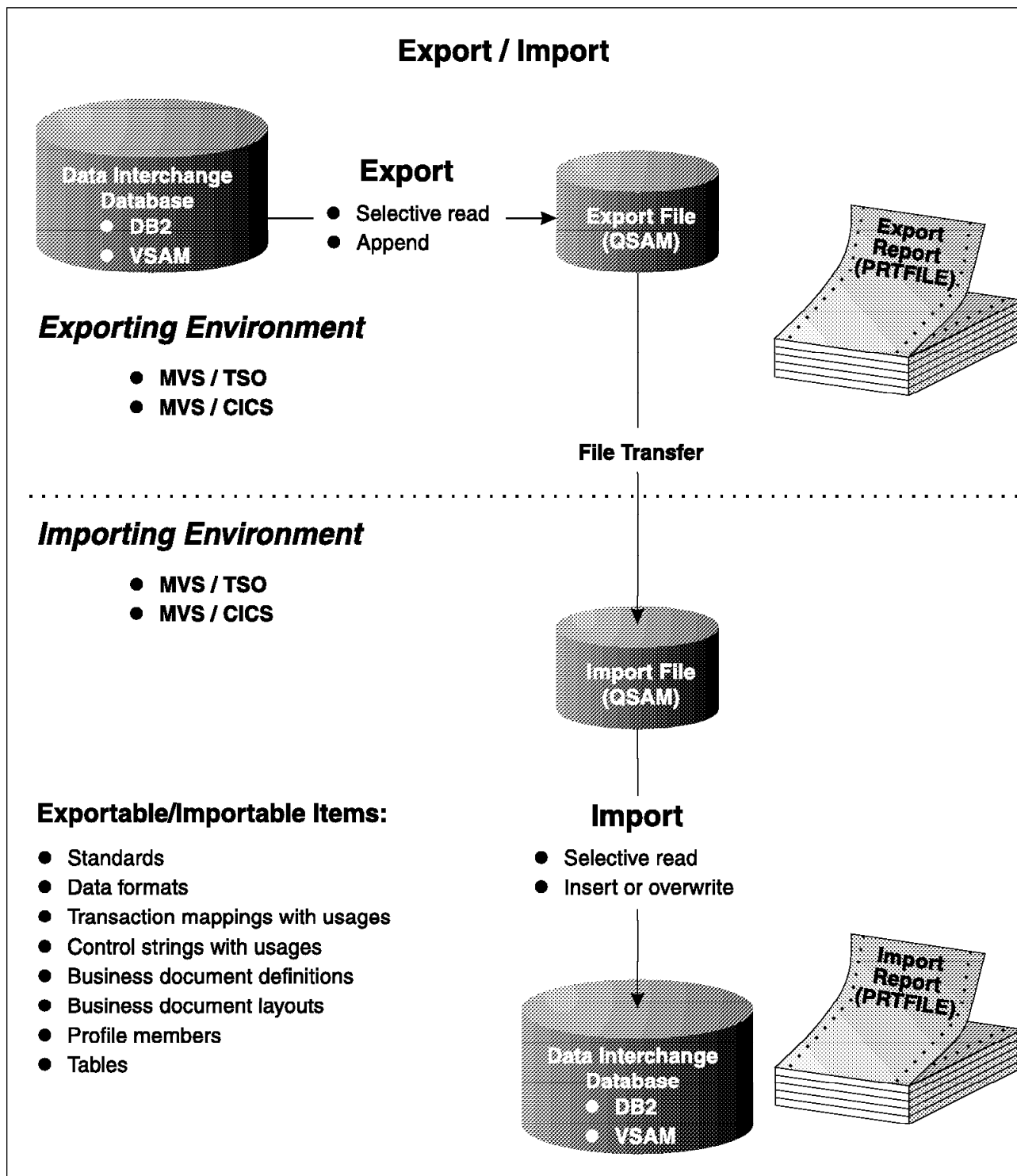


Figure 11-1. The Export/Import Process

You can request export and import interactively, using the *Export* and *Import* options on the Administrator's Menu (MP01), or you can use a utility command.

Export and Import supports two different file record formats. The TAGGED format puts all of the data into tag form. The FIXED format places the data into fixed positions for a fixed length. For online export, if the file is empty, you are prompted for the desired format. For the utility export, you can specify the requested format with the EIFORMAT keyword.

Note: You must use the tagged format when exchanging data between different releases of DataInterchange.

This chapter describes the interactive export/import methods. The utility commands are described in the *DataInterchange Programmer's Reference*.

Export and Import File List

When using export or import, you can create one data set and associate each of the following data definition names (ddnames) to the data set, or you can create a different data set for each ddname (see the *EIFILE* option in “Logon Options for DataInterchange for MVS” on page 2-1). For DataInterchange for CICS, the export files are transient data queues (TDQs).

Category	DDname(TSO)	TDQ(CICS)
Standards	EDIEISTD	EDIS
Application data formats	EDIEIADF	EDIA
Trading partner transactions	EDIEITPT	EDIT
Control strings	EDIEICST	EDIC
Business document definitions	EDIEIDDF	EDID
Business document layouts	EDIEIMAP	EDIM
Profiles	EDIEIPRF	EDIP
Tables	EDIEITBL	EDIB

Note: See the *DataInterchange Programmer's Reference* for a description of these files.

Note: Allocating all of the ddnames to a single physical data set can slow the processing of large files.

The *DataInterchange Programmer's Reference* describes how to manage the export and import files.

Export and Import Restrictions and Limitations

Export and import have the following restrictions:

- When you move data from one environment to another, language translation does not occur. English data remains English data.
- The import and export environments on both systems must be at compatible release and maintenance levels.
- You must have the authority to access the export and import menu options and to work with the objects to be exported or imported (see Appendix A, “Security”).
- When importing associated objects, the imported objects replace duplicate entries in DataInterchange without warning. For example, if you import a trading partner transaction with a network profile member as an associated object, the network profile member will replace any existing member with the same name.
- The field position and length of fixed format records are subject to change due to design changes within DataInterchange. When fields are added to records, attempts are made to add them to the end of the record to lessen the impact.

- You must use TAGGED format when importing between two different releases; for example, if you are at release 2.1 and want to import data from a trading partner that is at release 1.5, your trading partner must export the data in TAGGED format before you will be able to import it.

Using the Export Menu (EI00) Options

When you select *Export* from the Administrator's Menu (MP01), the Export Menu (EI00) is displayed.

```

EI00                                Export Menu

Type the number of your selection below and press Enter, or press the exit
key to exit.

Choice ==> _  1. Standards
               2. Application data formats
               3. Trading partner transactions
               4. Control strings
               5. Business document definitions
               6. Business document layouts
               7. Profiles
               8. Tables

```

This menu lists each type of administrative data, or primary object, that you can export as a separate menu option. Each primary object may have associated objects that can be exported with the primary object. Remember that only primary objects that have been exported can be selected for import, and all associated objects exported with the primary object are automatically imported when the primary object is imported.

The following table describes the options on the Export Menu (EI00).

Option	Exports
Standards	A complete standard or one or more transaction set definitions.
Application data formats	A data format.
Trading partner transactions	A transaction mapping and/or trading partner usages.
Control strings	A translation control string for a transaction mapping.
Business document definitions	A business document definition used by the Interactive Entry Facility.
Business document layouts	A business document layout used by the Interactive Entry Facility.
Profiles	All or selected members of a profile.
Tables	A translation or validation table.

After you press Enter on the menu panel, if the file associated with the object being exported is empty, panel EI31 is displayed to prompt you for the file record format.

Exporting Standards

You can export an entire standard or specific transaction sets within a standard. The complete standard includes:

- Standard definition
- All transaction sets in the standard
- All segments in the standard
- All segment usages in the standard
- All data elements in the standard
- All data element usages in the standard

A transaction set includes:

- Standard definition
- Transaction set definition
- All segments in the transaction set
- All segment usages in the transaction set
- All data elements in the transaction set
- All data element usages in the transaction set

You can also export associated objects for a standard, such as:

- Envelope standard
- Envelope profile
- Validation tables

To export standards, follow these steps:

1. Select *Export* from the Administrator's Menu (MP01). The Export Menu (EI00) is displayed.
2. Select *Standards*. The Export Standards panel (EI02) is displayed, listing the standards on your system that you can export.
3. Type **E** in the action column next to the standard you want to export, and press Enter.

Export List				
EI02		Export Standards		1 to 3 of 3
A	Standard ID	Description	Vers	Rel
e	EDI902	UN/EDIFACT Standard (1990 Release 2)	90	
—	TDCC	TDCC/UCS Standards (Version 2/6)	02	06
—	X12	ANSI X12 generic business format	02	02

The Associated Objects for Standards panel (EI03) is displayed.

EI02		Export Standards	1 to 3 of 3	
A	Standard ID	Description	Vers	Rel
e	MYEDI902	UN/EDIFACT Standard (1990 Release 2)	90	02
-	TDCC	TDCC/UCS Standards (Version 2/6)	02	06
-	X12			

EI03		Associated Objects for Standards
Standard ID		EI902
Envelope type		E
All transaction sets (Y/N) . .		N
Validation tables (Y/N) . . .		N
Envelope profile (Y/N)		N
Envelope standard (Y/N) . . .		N

4. Complete the fields as follows:

In this field:

All transaction sets

Enter:

Y if you want to export the entire standard, including all data elements, segments, and transaction set definitions, or **N** if you want to export only selected transaction set definitions.

Note: Transaction sets are not associated objects; however, you must tell DataInterchange if you want to select specific transaction sets.

Validation tables

Y if you want to export the validation tables associated with the standard or the selected transaction sets, or **N** if you do not.

Envelope profile

Y if you want to export the envelope profile associated with the standard definition, or **N** if you do not.

Envelope standard

Y if you want to export the envelope definitions associated with the standard definition, or **N** if you do not.

5. Press Enter.

If you specified N in the *All transaction sets* field, the Standard Transaction Sets panel (EI04) is displayed, listing the transaction sets defined in the standard you selected.

6. Type **E** in the action column next to each transaction set you want to export, and press Enter.

Export List		
EI04		Standard Transaction Sets 1 to 12 of 19
Standard ID : MYEDI902		
Action	Trans ID	Description
	CONTRL	Acknowledgment/Rejection Advice Message
e	CREADV	Credit Advice Message
-	CREEXT	Extended credit advice
-	CUSDEC	Customs Declaration Message
-	CUSRES	Customs Response Message
-	DEBADV	Debit Advice Message
-	IFTMAN	Arrival Notice
-	IFTMBC	Booking Confirmation
-	IFTMBF	Firm Booking
-	IFTMBP	Provisional Booking
e	IFTMCS	Instruction Contract Status
-	IFTMFR	International FWDing and Transport Msg Framework

DataInterchange writes the standard or selected transaction sets and the associated objects you specified to the appropriate export file, then redisplay the Export Standards panel (EI02).

7. Press F3 (Exit) twice to return to the Administrator's Menu (MP01).

Exporting Application Data Formats

To export specific application data formats, follow these steps:

1. Select *Export* from the Administrator's Menu (MP01). The Export Menu (EI00) is displayed.
2. Select *Application data formats*. The Export Application Data Formats panel (EI05) is displayed, listing the application data formats that are defined on your system.
3. Type **E** in the action column next to each application data format you want to export, and press Enter.

Export List		
EI05 Export Application Data Formats 1 to 4 of 4		
Action	Data Format ID	Description
e	GENINV	General Use Invoice - Send
-	GENINRCV	General Use Invoice - Receive
-	GENPO	General Use Purchase Order - Send
-	GENPORCV	General Use Purchase Order - Send

DataInterchange writes the application data formats to the appropriate export file.

4. Press F3 (Exit) twice to return to the Administrator's Menu (MP01).

Exporting Trading Partner Transactions (Maps)

You can export an entire trading partner transaction, or only the trading partner usages. The entire trading partner transaction includes:

- Trading partner transaction definition
- All segments used in the trading partner transaction
- All data elements used in the trading partner transaction

You can also export the following objects associated with a trading partner transaction:

- Mapping
- Trading partner usages
- Control string
- Standard transaction set
- Application data format
- Validation tables
- Translation tables
- User exit routines
- Trading partner profile
- Translation exit routines
- Security profile
- Network profile
- Envelope profile
- Network operations profile
- Envelope standard

To export trading partner transactions, follow these steps:

1. Select *Export* from the Administrator's Menu (MP01). The Export Menu (EI00) is displayed.
2. Select *Trading partner transactions*. The Export TP Transactions panel (EI06) is displayed, listing the trading partner transactions on your system that you can export.

Note: The TYPE column specifies whether the trading partner transaction was previously transferred from the host to the DataInterchange Client. If the value in the TYPE column is C, I, or P, the TPT was previously moved to the DataInterchange Client. It is questionable whether the host TPT should be exported, because the most recent copy of the map source is on the DataInterchange Client.

- C Indicates the host map was converted to a DataInterchange Client map database format. C also indicates the control string residing on the host was generated on the DataInterchange Client.
 - I Indicates the host map was converted to a DataInterchange Client map database format, but the host control string was generated using the host map.
 - P Indicates the host has only the map header; the complete map is stored in the DataInterchange Client map database. The control string, if any, was generated on the DataInterchange Client.
3. Type **E** in the action column next to the trading partner transaction you want to export, and press Enter.

Export List				
EI06		Export TP Transactions		1 to 4 of 4
A	Transaction ID	Send/Recv	Type	Description
e	GENINVMAP	S		Generic mapping for an invoice - send
-	GENINVRCVMAP	R		Generic mapping for an invoice - receive
-	GENPOMAP	S	C	Generic mapping for purchase orders - send
-	GENPORCVMAP	R	P	Generic purchase order mapping - receive

The Associated Objects for TP Transactions panel (EI07) is displayed.

Associated Objects for TP Transactions				
EI07				
A	T			
e	G	Transaction ID : GENINVMAP	Send/Receive : S	
-	G	Standard ID . . : X12V2R2	Std. Tran. ID : 810	
-	G	Data Format ID : GENINV		
Mapping (Y/N) Y+				
All TP usages (Y/N) N+ Control string (Y/N) N+				
Standard transaction (Y/N) . . N+ Application data format (Y/N) . . N+				
Validation tables (Y/N) N+ Translation tables (Y/N) N+				
User exit routines (Y/N) N+ Trading partner profile (Y/N) . . N+				
Translation exit routines (Y/N) N+ Security profile (Y/N) N+				
Network profile (Y/N) N+ Envelope profile (Y/N) N+				
Network operations (Y/N). . . . N+ Envelope standard (Y/N) N+				

4. Complete the fields as follows:

In this field:	Enter:
Mapping	Y to export the mapping associated with this trading partner transaction, or N to export only the trading partner usages.
All TP usages	Y to export all trading partner usages, or N to export only selected trading partner usages. If you specify N , the Trading Partner Usage for Sending panel (EI08) or the Trading Partner Usage for Receiving panel (EI09) is displayed with a list of usages from which you can select. If you do not want to export <i>any</i> usages, press Enter on the Trading Partner Usage for Sending panel (EI08) or the Trading Partner Usage for Receiving panel (EI09) without selecting any usage. You must press Enter.
Control string	Y if you want to export control string records for this trading partner transaction, or N if you do not. If you specify Y , <i>Mapping</i> must also be Y . The control string is the information the translator uses when translating the transaction between application and EDI format.
Standard transaction	Y if you want to export the standard transaction set associated with the trading partner transaction, or N if you do not.
Application data format	Y if you want to export the data format that is associated with this trading partner transaction, or N if you do not.
Validation tables	Y if you want to export the validation tables referred to by the trading partner transaction, or N if you do not.
Translation tables	Y if you want to export the translation tables referred to by the trading partner transaction, or N if you do not.
User exit routines	Y if you want to export the user program information (ADAMCTL) profile members whose names are the same as exit routine names that appear in the trading partner transaction, or N if you do not.
Trading partner profile	Y if you want to export the trading partner profile members that are associated with the selected trading partner usages, or N if you do not. When exporting generic send or receive usages, the trading partner profiles are not automatically exported. You must export the related profile members in a separate export.
Translation exit routines	Y if you want to export the user program information (ADAMCTL) profile members whose names are the same as post-translation (send) and pre-translation (receive) exit routine names that appear in the selected trading partner usages, or N if you do not.
Envelope profile	Y if you want to export the envelope profile members associated with the selected trading partner usages, or N if you do not. When exporting send or receive usages and a generic standard envelope profile is specified in the usage override, the envelope profile members are not automatically exported. You must export the related profile members in a separate export.

5. Press Enter.

If you specified N in the *All TP usages* field, a Trading Partner Usage panel is displayed with a list of usages from which you can select.

For a send transaction, the Trading Partner Usage for Sending panel (EI08) is displayed.

Export List	
EI08 Trading Partner Usage for Sending 1 to 2 of 2	
Transaction ID : GENINVMAP	
A	Ref Sender's Receiver's Receiver's Num TP Nickname TP Nickname Internal Trading Partner ID 001 CAPRICORN CAPRICORN TUTORPARTNER e 002 PISCES PISCES AAAAAA T A y c p t e Y Y Y

For a receive transaction, the Trading Partner Usage for Receiving panel (EI09) is displayed.

6. Type **E** in the action column next to each trading partner usage you want to export, and press Enter.

Export List	
EI09 Trading Partner Usage for Receiving 1 to 3 of 3	
Transaction ID : GENINVRCVMAP	
A	Ref Trading Partner Application Sender(S) or Receiver(R) ID Active Test Num Nickname 001 ARIES DEPT130 (S) Y N 002 CAPRICORN DEPT130 (S) Y Y e 003 PISCES DEPT130 (R) N

DataInterchange writes the mapping or specified usages, or both, and all the associated objects you specified to the appropriate export file, then redisplayes the Export TP Transactions panel (EI06).

7. Press F3 (Exit) twice to return to the Administrator's Menu (MP01).

Exporting Control Strings

A *control string* is a set of instructions created with the *Generate* action after mapping a transaction. The control string guides the translator as it translates your data to standard format or standard data to your local format.

Exporting the control string includes:

- Control string
- Trading partner transaction definition
- Standard definition
- Application data format

You can also export the following objects associated with a trading partner transaction:

- Trading partner usages
- Validation tables
- Translation tables
- User exit routines

- Trading partner profile
- Translation exit routines
- Envelope profile

The advantage in exporting the control string, as opposed to the transaction mapping, is that it requires less space. The disadvantage is that you cannot regenerate the control string. You would export a control string only after you have customized and tested it.

To export control strings, follow these steps:

1. Select *Export* from the Administrator's Menu (MP01). The Export Menu (EI00) is displayed.
2. Select *Control strings*. The Export Control Strings panel (EI10) is displayed, listing all the translation control strings that could be generated. The TYPE column specifies whether the trading partner transaction was previously transferred from the host to the DataInterchange Client. If the value in the TYPE column is P or C, the Trading Partner Transaction was previously moved to the DataInterchange Client. Carefully determine if this is the control string you really want to export.
3. Type **E** in the action column next to the control string you want to export, and press Enter.

Export List			
EI10		Export Control Strings	
		1 to 1 of 4	
A	Transaction ID	Send/Recv	Type Description
e	GENINVMAP	S	Generic mapping for an invoice - send
-	GENINVRVMAP	R	Generic mapping for an invoice - receive
-	GENPOMAP	S	C Generic mapping for purchase orders - send
-	GENPORCVMAP	R	P Generic purchase order mapping - receive

The Associated Objects for Control Strings panel (EI11) is displayed.

```

A T
e G
- G
- G
- G
  EI11          Associated Objects for Control Strings
  Transaction ID . . . . . : GENINVMAP          Send/Receive . . : S
  All TP usages (Y/N) . . . . . N+
  Validation tables (Y/N) . . . . N+ Translation tables (Y/N) . . . N+
  User exit routines (Y/N) . . . . N+ Trading partner profile (Y/N) N+
  Translation exit routines (Y/N) N+ Envelope Profile (Y/N)          N+

```

4. Complete the fields as follows:

In this field:

All TP usages

Enter:

Y to export all trading partner usages, or **N** to export only selected trading partner usages. If you specify N, the Trading Partner Usage for Sending panel (EI08) or the Trading Partner Usage for Receiving panel (EI09) is displayed with a list of usages from which you can select.

Validation tables

Y if you want to export the validation tables referred to by the trading partner transaction, or **N** if you do not.

Translation tables

Y if you want to export the translation tables referred to by the trading partner transaction, or **N** if you do not.

In this field:	Enter:
User exit routines	Y if you want to export the user program information (ADAMCTL) profile members whose names are the same as exit routine names that appear in the trading partner transaction, or N if you do not.
Trading partner profile	Y if you want to export the trading partner profile members that are associated with the selected trading partner usages, or N if you do not.
Translation exit routines	Y if you want to export the user program information (ADAMCTL) profile members whose names are the same as post-translation (send) and pre-translation (receive) exit routine names that appear in the selected trading partner usages, or N if you do not.
Envelope profile	Y if you want to export the envelope profile members associated with the selected trading partner usages, or N if you do not.

5. Press Enter.

If you specified N in the *All TP usages* field, the Trading Partner Usage for Sending panel (EI08) or the Trading Partner Usage for Receiving panel (EI09) is displayed with a list of usages from which you can select. The Trading Partner Usage for Sending panel (EI08) is shown on page 11-10. The Trading Partner Usage for Receiving panel (EI09) is shown on page 11-10.

6. Type **E** in the action column next to each trading partner usage you want to export, and press Enter.

DataInterchange writes the control string and all the associated objects you specified to the appropriate export file, then redisplay the Export Control Strings panel (EI10).

7. Press F3 (Exit) twice to return to the Administrator's Menu (MP01).

Exporting Business Document Definitions

A *business document definition* establishes a connection between an Interactive Entry Facility document and an application data format. It can also provide additional information, such as the name of the requestor that sends or receives the document. In addition to the business document definition, you can export the application data format, business document layout, and requestor profile member as associated objects.

To export business document definitions, follow these steps:

1. Select *Export* from the Administrator's Menu (MP01). The Export Menu (EI00) is displayed.
2. Select *Business document definitions*. The Export Business Document Definitions panel (EI12) is displayed, listing the business document definitions that are known to the Interactive Entry Facility.

3. Type **E** in the action column next to the business document definition you want to export, and press Enter.

Export List		
EI12	Export Business Document Definitions	1 to 4 of 4
A	Data Format ID	Description
e	GENINV	General Use Invoice - Send
-	GENINRCV	General Use Invoice - Receive
-	GENPO	General Use Purchase Order - Send
-	GENPORCV	General Use Purchase Order - Receive

The Associated Objects for Document Definitions panel (EI13) is displayed.

EI12	Export Business Document Definitions	1 to 4 of 4
Action	Data Format ID	Description
e	GENINV	General Use Invoice - Send
-		
-		
-		
	<div> <div>EI13</div> <div>Associated Objects for Document Definitions</div> <div> Data Format ID : GENINV Requestor ID : REQABC Application data format (Y/N) . . N Business document layout (Y/N) . N Requestor profile (Y/N) N </div> </div>	

4. Complete the fields as follows:

In this field:

Application data format

Business document layout

Requestor profile

Enter:

Y if you want to export the application data format associated with the business document definition, or **N** if you do not.

Y if you want to export all business document layouts associated with the business document definition, or **N** if you do not.

Y if you want to export the requestor profile member associated with the business document definition, or **N** if you do not.

Note: If you have not associated a requestor ID with the business document definition (*Requestor ID* is blank), this field is not displayed.

5. Press Enter.

DataInterchange writes the business document definition and all the associated objects you specified to the appropriate export file, then redisplay the Export Business Document Definition panel (EI12).

6. Press F3 (Exit) twice to return to the Administrator's Menu (MP01).

Exporting Business Document Layouts

A *document layout* describes the appearance of the document when viewed or printed using the Interactive Entry Facility. You can export the associated application data format along with a document layout.

To export document layouts, follow these steps:

1. Select *Export* from the Administrator's Menu (MP01). The Export Menu (EI00) is displayed.
2. Select *Business document layouts*. The Export Business Document Layouts panel (EI14) is displayed, listing the application data formats for business document layouts that are defined to the Interactive Entry Facility.
3. Type **E** in the action column next to the document layout you want to export, and press Enter.

Export List		
<hr/>		
EI14	Export Business Document Layouts	1 to 4 of 4
A	Data Format ID	Description
e	GENINV	General Use Invoice - Send
-	GENINVR	General Use Invoice - Receive
-	GENPO	General Use Purchase Order - Send
-	GENPORCV	General Use Purchase Order - Receive

The Associated Objects for Document Layouts panel (EI15) panel is displayed.

<hr/>		
EI14	Export Business Document Layouts	1 to 3 of 3
Action	Data Format ID	Description
e	GENINV	General Use Invoice - Send
-		
-	<div><div>EI15</div><div>Associated Objects for Document Layouts</div><div>Data Format ID : GENINV</div><div>Application data format (Y/N) . . N+</div></div>	

4. Type **Y** in the *Application data format* field if you want to export the application data format identified on the panel, or type **N** if you do not.

DataInterchange writes the business document and the application data format, if specified, to the appropriate export file, then redisplay the Export Business Document Layouts panel (EI14).

5. Press F3 (Exit) twice to return to the Administrator's Menu (MP01).

Exporting Profiles

You can select which profiles you want to export. For a given profile, you can export all the members or selected members.

To export profiles, follow these steps:

1. Select *Export* from the Administrator's Menu (MP01). The Export Menu (EI00) is displayed.

2. Select *Profiles*. The Export Profiles panel (EI23) is displayed, listing the profiles that are defined for your system.
3. Type **E** in the action column next to the profile you want to export, and press Enter.

Export		
EI23		
Export Profiles		
1 to 15 of 17		
Action	Profile ID	Description
-	ACTLOGS	Activity log
-	ADAMCTL	User program information
-	APPDEFS	Application Definition Profile
-	CONTRECV	Continuous Receive Profile
-	E	EDIFACT standard envelope data
-	I	ICS standard envelope data
-	LANGPROF	Language profile
-	MQSERIES	MQSeries Queue Profile
-	NETOP	Network operation profile
-	NETPROF	Network profile
-	REQPROF	Requestor profile
-	SECUPROF	Security profile
-	SYSPROF	System profile
-	T	UN/TDI standard envelope data
e	TPPROF	Trading partner profile
-	U	UCS standard envelope data
-	X	X12 standard envelope data

The Profile Members panel (EI24) is displayed, listing the members of the profile you selected.

Export List		
EI23		
Export Profiles		
1 to 17 of 17		
Action	Profile ID	
-	ACTLOGS	
-	ADAMCTL	
-	CONTRECV	
-	E	
-	I	
-	LANGPROF	
-	MQSERIES	
-	NETOP	
-	NETPROF	
-	REQPROF	
-	SECUPROF	
-	SYSPROF	
-	T	UN/TDI standard envelope data
e	TPPROF	Trading Partner profile
-	U	UCS standard envelope data
-	X	X12 standard envelope data

EI24		
Profile Members		
1 to 3 of 3		
Profile ID : TPPROF		
Export all of this profile's members? (Y/N) N		
-	ARIES	
-	CAPRICORN	
e	PISCES	

4. To export all members of this profile, type **Y** in the *Export all of this profile's members?* field, and press Enter.

To export selected members of this profile:

- a. Type **N** in the *Export all of this profile's members?* field.
- b. Type **E** in the action column next to each member you want to export.
- c. Press Enter.

DataInterchange writes the specified members to the appropriate export file, then redisplay the Export Profiles panel (EI23).

5. Press F3 (Exit) twice to return to the Administrator's Menu (MP01).

Exporting Tables

To export specific translation or validation tables, follow these steps:

1. Select *Export* from the Administrator's Menu (MP01). The Export Menu (EI00) is displayed.
2. Select *Tables*. The Export Tables panel (EI27) is displayed, listing the tables that are defined for your system.
3. Type an **E** in the action column next to the table you want to export, and press Enter.

Export List			
EI27			
Export Tables			
1 to 17 of 500			
Action	Table ID	Type	Description
e	ALPHANUM	V	Alphanumeric validation table
	CHARSET	V	Character set validation table
	CLAS	V	Order Classification
	CRLI	V	Credit Line Indicator
	CSDI	V	Cash Settlement Discount Identifier
	DELC	V	Delivery Condition Codes
	DIEDTMSK	T	EDIFACT Date Masks
	DIMONNUM	T	Numeric month to textual month
	DIMONTXT	T	Textual month to numeric month
	DIXDTMSK	T	X12 Date Masks
	DI14REG	T	Translate Tabel for DI 1.4 Regr Tst
	FILENAME	V	File name validation table
	INV374	T	Date Qualifier for GENINVRCV
	INV560	T	Service Code for GENINVRCV
	INV98	T	Entity ID Code for GENINVRCV
	LINE	V	Statement/Remittance Line Code
	MONOCASE	T	Monocasing translate table

DataInterchange writes the specified tables to the appropriate export file, then redisplayes the Export Tables panel (EI27).

4. Press F3 (Exit) twice to return to the Administrator's Menu (MP01).

Using the Import Menu (EI01) Options

When you select *Import* from the Administrator's Menu (MP01), the Import Menu (EI01) is displayed.

EI01	Import Menu
Type the number of your selection below and press Enter, or press the Exit key to exit.	
Choice ==> _	1. Standards
	2. Application data formats
	3. Trading partner transactions
	4. Control strings
	5. Business document definitions
	6. Business document layouts
	7. Profiles
	8. Tables

This menu lists each type of administrative data, or primary object, that you can import. Each primary object is a separate menu option. Some primary objects might have associated objects which are imported along with the primary object. You cannot import the associated objects without the primary object. The following table describes the options on the Import Menu (EI01).

Option	Imports:
Standards	A complete standard or one or more transaction set definitions.
Application data formats	A data format.
Trading partner transactions	A transaction mapping and/or trading partner usages.
Control strings	A translation control string for a transaction mapping. You can include all or selected trading partner usages.
Business document definitions	A business document definition used by the Interactive Entry Facility.
Business document layouts	A business document layout used by the Interactive Entry Facility.
Profiles	All or selected members of a profile.
Tables	A translation or validation table.

Importing Standards

To import standards, follow these steps:

1. Select *Import* from the Administrator's Menu (MP01). The Import Menu (EI01) is displayed.
2. Select *Standards*. The Import Standards panel (EI16) is displayed, listing the standards that are in your import file.
3. Type **I** next to the standard you want to import, and press Enter.

Import					
EI16			Import Standards		1 to 3 of 3
A	Standard ID	Description	Vers	Rel	
—	MYEDI902	UN/EDIFACT Standard (1990 Release 2)	90		
—	TDCC	TDCC/UCS Standards (Version 2/6)	02	06	
—	X12	ANSI X12 generic business format	02	02	
i	X003042	ANSI X12 Version 3, Rel 4, Sub 2, 003042 Part 1	03	42	

If a standard by that name already exists in the DataInterchange database, the Duplicate Object panel (EI22) is displayed.

EI16		Import Standards	1 to 3 of 3	
A	Standard ID	Description	Vers	Rel
-	MYEDI902	UN/EDIFACT Standard (1990 Release 2)	90	02
-	TDCC		02	
-	X12	EI22 Duplicate Object	02	02
-	X003042		03	42

W A R N I N G

Selected ID : X003042

The object you selected already exists in your DataInterchange data base. If you want to replace it, press Enter to continue. Or press Cancel to cancel the request.

- If you want to replace it, press Enter to continue, or press F12 (Cancel) to cancel the request.

Import

EI32 Standard Transaction Sets 1 to 13 of 214

Standard ID : X003042 All Transactions (Y/N) : y +

Action	Trans ID	Description
-	104	Air Shipment Information
-	110	Air Freight Details and Invoice
-	120	Vehicle Shipping Order
-	121	Vehicle Service
-	125	Multilevel Railcar Load Details
-	126	Vehicle Application Advice
-	127	Vehicle Baying Order
-	128	Dealer Information
-	129	Vehicle Carrier Rate Update
-	130	Student Educational Record (Transcript)
-	131	Student Educational Record (Transcript) Acknow
-	135	Student Loan Application
-	139	Student Loan Guarantee Result

Command ==>

Enter Tso F1=Help F3=Exit F4=Prompt F8=Fwd F9=Retrieve

F12=Cancel F13=Keys help

The Standard Transaction Sets panel (EI32) is displayed.

- If you want to import all transaction sets, type **Y** in the *All Transactions* field. If you want to select a specific transaction set, type **N** in the *All Transactions* field, type **I** next to the standard transaction set you want to import, and press Enter.

DataInterchange imports the standard or standard transaction sets selected, and then redisplay the Standard Transaction Sets panel (EI32).

- Press F3 (Exit) three times to return to the Administrator's Menu (MP01).

Importing Application Data Formats

To import application data formats, follow these steps:

- Select *Import* from the Administrator's Menu (MP01). The Import Menu (EI01) is displayed.
- Select *Application data formats*. The Import Application Data Formats panel (EI17) is displayed, listing the application data formats that are in your import file.

3. Type **I** next to the application data format you want to import, and press Enter.

Import		
EI17 Import Application Data Formats 1 to 4 of 4		
Action	Data Format ID	Description
i	GENINV	General Use Invoice - Send
-	GENINVRVCV	General Use Invoice - Receive
-	GENPO	General Use Purchase Order - Send
-	GENPORCV	General Use Purchase Order - Receive

If an application data format by that name already exists in the DataInterchange database, the Duplicate Object panel (EI22) is displayed.

4. If you want to replace the application data format, press Enter to continue, or press F12 (Cancel) to cancel the request.

DataInterchange imports the application data format, then redisplay the Import Application Data Formats panel (EI17).

5. Press F3 (Exit) twice to return to the Administrator's Menu (MP01).

Importing Trading Partner Transactions (Maps)

To import trading partner transactions, follow these steps:

1. Select *Import* from the Administrator's Menu (MP01). The Import Menu (EI01) is displayed.
2. Select *Trading partner transactions*. The Activate Usage panel (EI29) is displayed to prompt for the activation option when importing TP usages. Following are the activation options:
 - Do not replace existing active usage.
 - Replace the existing usage with active imported usage.
 - Force all imported usages to be active.

3. Select the desired option and press Enter. The Import TP Transactions panel (EI18) is displayed, listing the trading partner transactions that are in the import file.

Note: A **Y** in the *Usage Only?* column indicates that only the trading partner usages were exported for the transaction.

4. Type **I** next to the trading partner transaction you want to import, and press Enter.

Note: The import of active usage for a trading partner with an existing active usage may deactivate the existing active usage. The options for importing active usages are selected from panel EI29.

Import				
EI18 Import TP Transactions 1 to 4 of 4				
A	Transaction ID	Send/Recv	Description	Usage Only?
	GENINVMAP	S	Generic mapping for invoice - send	N
i	GENINVRVCV	R	Generic mapping for invoice - receive	
-	GENPOMAP	S	Generic mapping for purchase order - send	Y
-	GENPORCV	R	Generic mapping for purchase order - receive	N

If the trading partner transaction to be imported includes a transaction mapping, and that mapping already exists in the DataInterchange database, the Duplicate Object panel (EI22) is displayed.

- If you want to replace the transaction, press Enter to continue, or press F12 (Cancel) to cancel the request.

If the transaction has only trading partner usages, the TP Usage Association panel (EI30) is displayed.

EI18 Import TP Transactions				1 to 4 of 4
A	Transaction ID	Send/Recv	Description	Usage Only?
-	GENINVMAP			
-	GENINVRVCM	EI30	TP Usage Association	Y
-	GENPOMAP			Y
-	GENPORCVMA			N

The object that you have selected is an import of TP Usages only. To associate these usages with a transaction other than the one shown, type the TP transaction ID and press enter. To associate them with the one shown, simply press enter.

TP Transaction ID . . . GENINVRVCM_____

- To import the usages and associate them with the trading partner transaction shown, press Enter.
- To import the usages and associate them with a different trading partner transaction, type the transaction ID, and press Enter. The transaction must already exist in the DataInterchange database, and must be compatible with the usages. For example, if importing send usages, the transaction you specify must be a send transaction.

DataInterchange imports the trading partner transaction or usage, and then redisplay the Import TP Transactions panel (EI18).

- Press F3 (Exit) twice to return to the Administrator's Menu (MP01).

Importing Control Strings

To import control strings, follow these steps:

- Select *Import* from the Administrator's Menu (MP01). The Import Menu (EI01) is displayed.
- Select *Control strings*. The Activate Usage panel (EI29) is displayed to prompt for the activation option when importing TP usages.
- Select the desired option and press Enter. The Import TP Transactions panel (EI19) is displayed, listing the control strings that are in the import file.

4. Type **I** next to the control string you want to import, and press Enter.

Import		
EI19		Import Control Strings
		1 to 4 of 4
Action	Transaction ID	
i	GENINVMAP	
-	GENINVRCVMAP	
-	GENPOMAP	
-	GENPORCVMAP	

If a control string for that transaction already exists in the DataInterchange database, the Duplicate Object panel (EI22) is displayed.

5. If you want to replace the control string, press Enter to continue, or press F12 (Cancel) to cancel the request.

DataInterchange imports the control string, then redisplay the Import Control String panel (EI19).

6. Press F3 (Exit) twice to return to the Administrator's Menu (MP01).

Importing Business Document Definitions

To import business document definitions, follow these steps:

1. Select *Import* from the Administrator's Menu (MP01). The Import Menu (EI01) is displayed.
2. Select *Business document definitions*. The Import Business Document Definitions panel (EI20) is displayed, listing the business document definitions that are in your import file.
3. Type **I** next to the business document definition you want to import, and press Enter.

Import		
EI20		Import Business Document Definitions
		1 to 4 of 4
Action	Data Format ID	
i	GENINV	
-	GENINVRCV	
-	GENPO	
-	GENPORCV	

If the business document definition already exists in the DataInterchange database, the Duplicate Object panel (EI22) is displayed.

4. If you want to replace the business document definition, press Enter to continue, or press F12 (Cancel) to cancel the request.

DataInterchange imports the business document definition, then redisplay the Import Business Document Definitions panel (EI20).

5. Press F3 (Exit) twice to return to the Administrator's Menu (MP01).

Importing Business Document Layouts

To import business document layouts, follow these steps:

1. Select *Import* from the Administrator's Menu (MP01). The Import Menu (EI01) is displayed.
2. Select *Business document layouts*. The Import Business Document Layouts panel (EI21) is displayed, listing the document layouts that are in your import file.
3. Type **I** in the action column next to the document layouts you want to import, and press Enter.

Import		
<hr/>		
EI21	Import Business Document Layouts	1 to 3 of 3
Action	Data Format ID	
i	GENINV	
-	GENINVR	
-	GENPO	
-	GENPORCV	

If the business document layout already exists in the DataInterchange database, the Duplicate Object panel (EI22) is displayed.

4. If you want to replace the document layout, press Enter to continue, or press F12 (Cancel) to cancel the request.

DataInterchange imports the business document layout, then redisplay the Import Business Document Layout panel (EI21).

5. Press F3 (Exit) twice to return to the Administrator's Menu (MP01).

Importing Profiles

To import profiles, follow these steps:

1. Select *Import* from the Administrator's Menu (MP01). The Import Menu (EI01) is displayed.
2. Select *Profiles*. The Import Profiles panel (EI25) is displayed, listing the profiles that are in your import file.
3. Type **I** next to the profile type you want to import, and press Enter.

Import		
<hr/>		
EI25	Import Profiles	1 to 3 of 3
Action	Profile ID	Description
-	NETOP	Network operation profile
-	NETPROF	Network profile
i	TPPROF	Trading partner profile

The Profile Members panel (EI26) is displayed, listing the members of the profile you selected.

Import List	
EI25	Import Profiles 1 to 3 of 3
Action	Profile ID
-	NETOP
-	NETPROF
I	TPPROF

EI26 Profile Members 1 to 5 of 5	
Profile ID : TPPROF	
Import all of this profile's members? (Y/N) Y +	
Import TPPROF Control Number Values? (Y/N) Y +	
Action	Key
-	Company A
-	Company B
-	Company C
-	Company D
-	Company E

4. To import all members of this profile, type **Y** in the *Import all of this profile's members* field. To set the trading partner profile interchange, group, and Control Number values to those of the importing members, type **Y** in the *Import TPPROF Control Number Values* field. To retain values of existing TP profile members, type **N** in the field. Press Enter to begin importing.
5. To import selected members of this profile:
 - a. Type **N** in the *Import all of this profile's members* field.
 - b. Type **Y** or **N** in the *Import TPPROF Control Number Values* field.
 - c. Type **I** in the action column next to each member you want to import.
 - d. Press Enter.

If the profile members already exist in the DataInterchange database, the Duplicate Profile Member panel (EI39) is displayed.

6. If you want to replace the profile members, press Enter to continue, or press F12 (Cancel) to cancel the request. When importing trading partner profile members, DataInterchange generates error messages if you try to import a member whose:
 - Account number and user ID are duplicates of an existing member
 - Interchange ID and qualifier are duplicates of an existing member

DataInterchange imports the profile members, then redisplay the Import Profiles panel (EI25).

7. Press F3 (Exit) twice to return to the Administrator's Menu (MP01).

Importing Tables

To import translation or validation tables, follow these steps:

1. Select *Import* from the Administrator's Menu (MP01). The Import Menu (EI01) is displayed.
2. Select *Tables*. The Import Tables panel (EI28) is displayed, listing the tables that are defined for your system.

3. Type an **I** in the action column next to the table you want to import, and press Enter.

Import			
EI28		Import Tables	1 to 17 of 17
Action	Table ID	Type	Description
I	ALPHANUM	V	Alphanumeric validation table
	CHARSET	V	Character set validation table
	CLAS	V	Order Classification
	CRLI	V	Credit Line Indicator
	CSDI	V	Cash Settlement Discount Identifier
	DELC	V	Delivery Condition Codes
	DIEDTMSK	T	EDIFACT Date Masks
	DIMONNUM	T	Numeric month to textual month
	DIMONTXT	T	Textual month to numeric month
	DIXDTMSK	T	X12 Date Masks
	DI14REG	T	Translate Tabel for DI 1.4 Regr Tst
	FILENAME	V	File name validation table
	INV374	T	Date Qualifier for GENINVRV
	INV560	T	Service Code for GENINVRV
	INV98	T	Entity ID Code for GENINVRV
	LINE	V	Statement/Remittance Line Code
	MONOCASE	T	Monocasing translate table

DataInterchange imports the specified tables, then redisplay the Import Tables panel (EI28).

4. Press F3 (Exit) twice to return to the Administrator's Menu (MP01).

Printed Report

The export and import functions report the results of the processing in a file (PRTFILE) that you can review on your screen or print as hard copy using system facilities (see "Printing Reports and Lists" on page 2-11). For interactive processing, the titles of the reports are "DataInterchange Export" and "DataInterchange Import."

The export and import reports contain the following information:

- Date and time of export or import
- Selected object names
- Warning messages, such as DUPLICATE OBJECT REPLACED ON IMPORT
- Error messages, such as MISSING ASSOCIATED OBJECTS ON EXPORT
- Completion messages, such as YOUR REQUEST WAS COMPLETED SUCCESSFULLY

The following figures show examples of the export and import reports.

1	-DataInterchange Export-	Date: 97/12/14	Time: 10:57:00	Page
***** Exporting Standard EDI902 *****				
Your request was completed successfully				
***** Exporting TP usages from YMPEPOS *****				
Member EDI912E does not exist in profile E.				
Some associated objects were not processed--see report for details				
***** Exporting Profile TPPROF *****				
Your request was completed successfully				

Figure 11-2. Sample Report - DataInterchange Export

```

1          -DataInterchange Import-      Date: 97/12/14   Time: 11:08:40   Page

***** Importing Transaction YMPTPOS *****
Transaction YMPTPOS has been replaced in the data base.
Your request was completed successfully

***** Importing TP Usages to YMPEPOS *****
Your request was completed successfully

***** Importing Control string YMP823R *****
Control string MP YMP823R has been replaced in the data base.
Transaction YMP823R has been replaced in the data base.
Data format YAF823 has been replaced in the data base.
Your request was completed successfully

***** Importing Layout GENINV *****
Your request was completed successfully

***** Importing Profile TPPROF *****
Member COMPANYA has been replaced in profile TPPROF.
Member PARTNER has been replaced in profile TPPROF.
Your request was completed successfully

```

Figure 11-3. Sample Report - DataInterchange Import

Export/Import Facility and Utility Comparisons

The following export and import functions vary between the interactive and utility interfaces to export and import setup data.

Consider the difference between these functions when selecting the interactive or utility interface to export and import setup data.

Data	Interactive Interface	Utility Interface
Profiles	Exports the entire profile group or selected members of the profile group	Exports the entire profile group or a <i>single</i> selected member of the profile group
Standards	Exports the entire data group or selected members of the data group	Exports the selected members of a data group only
Application data formats		
Trading partner transactions		
Control strings		
Business document definitions		
Business document layouts		
Tables		

Chapter 12. Event Logging

Understanding Log Profiles	12-1
Activity Log Profile	12-1
Printing or Viewing Log Entries	12-4
Interpreting Log Entries	12-7
Translator Messages	12-7
Turning Logging On and Off	12-8
Defining an Event Log	12-9

Chapter 12. Event Logging

An event log is a record of activities that occur when you request DataInterchange services.

Events written to the log include:

- Program and database errors

When an error occurs, its symptoms are recorded to help support personnel identify and correct the problem.

- Network status

This type includes events such as queued for sending, sent, received, and delivered.

- Profile accesses

Gaining access to a profile produces a record that includes the user's ID and the name of the profile.

DataInterchange provides two event logs: LOGFFS and LOGEDI.

LOGFFS records events that occur when you use DataInterchange Utility services. Although LOGFFS is the default log, your request can specify another log.

LOGEDI records events that occur when you use any other DataInterchange services, such as the Transaction Store Facility or Interactive Entry Facility (IEF).

When working with event logs, you can view or print multiple event log entries; turn logging on and off for different events, such as for profiles access; add event logs for individual applications; and archive log entries by placing them in a history file.

Understanding Log Profiles

- | This section describes the activity log profile (ACTLOGS), which identifies applications and the event logs used for recording events associated with those applications.

Activity Log Profile

The activity log profile tells DataInterchange where to record events, such as sending or receiving a transaction that pertains to a business application. If you want each business application, such as purchasing and accounts receivable, to have its own event log file, add one member for each business application. If you want your business applications to share an event log file, add only one member. Applications that use the DataInterchange Utility to request translation and network services can share the log file LOGFFS supplied by DataInterchange, or you can define separate log files. See "Defining an Event Log" on page 12-9 for instructions.

To add a member to the activity log profile, follow these steps:

1. From the Administrator's Menu (MP01), select Profiles. The Profile Definitions panel (PM01) is displayed.

Event Logging

2. To view the profile definition before adding a member, type **V** in the action column next to ACTLOGS, and press Enter.

Log work with Members Print View			
PM01		Profile Definitions	1 to 17 of 17
Action	Profile ID	Profile Description	Log?
V	ACTLOGS	Activity log	N
-	ADAMCTL	User program information	N
-	APPDEFS	Application Definition Profile	N
-	CONTRECV	Continuous Receive Profile	N
-	E	EDIFACT standard envelope data	N
-	I	ICS standard envelope data	N
-	LANGPROF	Language profile	N
-	MQSERIES	MQSeries queue profile	N
-	NETOP	Network operation profile	N
-	NETPROF	Network profile	N
-	REQPROF	Requestor profile	N
-	SECUPROF	Security profile	N
-	SYSPROF	System profile	N
-	T	UN/TDI standard envelope data	N
-	TTPROF	Trading partner profile	Y
-	U	UCS standard envelope data	N
-	X	X12 standard envelope data	N

The View Profile Definition panel (PM05) is displayed.

PM05		View Profile Definition			1 to 3 of 3
Profile ID: ACTLOGS		Profile description: Activity log			
Key==>	Field label	Length	Type	Description	
	Application ID	008	CH	Application identification	
	Log file name	008	CH	Logical name of log file	
	Log flag	001	HX	Log flag ON (01) or OFF (00)	

The Field label column lists the prompts you see when providing data for a member of this profile. The first label is the key or name of this member. The next two columns show the characteristics of the data you can use. The last column contains a brief description of the fields.

3. Press F3 (Exit) when you are finished viewing the profile definition. The Profile Definitions panel (PM01) is redisplayed.
4. To add a new member or update an existing member, type **M** in the action column next to ACTLOGS, and press Enter. The Profile Members panel (PM07) is displayed.
5. Type **A** in the action column next to any item, and press Enter.

AddCopyDeleteListPrintUpdateView

PM01

Profile Definitions

1 to 17 of 17

ActionProfile ID

mACTLOGS

ADAMCTL

APPDEFS

CONTRECV

E

I

LANGPROF

MQSERIES

NETOP

NETPROF

REQPROF

SECUPROF

SYSPROF

T

TPPROF

U

X

PM07

Profile Members

1 to 2 of 2

Profile ID : ACTLOGS

Description: Activity log

ActionKey

aEDIFFS

EDIMP

MQSeries Queue Profile

N

Network operation profile

N

Network profile

N

Requestor profile

N

Security profile

N

System profile

N

UN/TDI standard envelope data

N

Trading partner profile

Y

UCS standard envelope data

N

X12 standard envelope data

N

The Add Profile Member panel (PM08) is displayed.

PM08

Add Profile Member

1 to 3 of 3

Profile ID: ACTLOGS

Fill in the information below and press Enter to save this member.

To stop entering members, press Exit or Cancel.

Application ID . .

Log file name . . .

Log flag 00

6. Complete the fields as follows:

In this field:
Application ID

Enter:
The name DataInterchange uses for this application. If your program uses the DataInterchange Utility to request translation and network services, the default name is EDIFFS. You can override the name with the APPLID keyword in the DataInterchange Utility parameters. If your program uses the application program interface to request services, the name must match the name of the program that calls DataInterchange with the initialization function code. This program name (the APPLID) can be switched dynamically after initialization by using the appropriate application program interface function.

Log file name

The data definition name (ddname) of the log file used by this application for recording events.

Applications that use the DataInterchange Utility can share the log file LOGFFS, which DataInterchange supplies. If you prefer, you can define and use a separate log for each application.

Log flag

01 to make logging active for this application, or **00** to make logging inactive.

Event Logging

7. Press Enter to save this information.
8. Press F3 (Exit) twice to return to the Administrator's Menu (MP01).

Printing or Viewing Log Entries

To print or view log entries, follow these steps:

1. From the Administrator's Menu (MP01), select Event Logging. The Event Logging panel (EL01) is displayed, listing the applications that have event logs.

Print	View
EL01	Event Logging 1 to 2 of 2
Action	Application ID
-	EDIFFS
-	EDIMP

EDIFFS is the application ID for the DataInterchange Utility, which uses the DataInterchange event log LOGFFS.

EDIMP is the application ID for the DataInterchange event log LOGEDI, which records entries for all DataInterchange services except the DataInterchange Utility, and for any of your applications that do not define an event log.

2. Type **P** or **V** in the action column next to the application you want to print or view log entries for, and press Enter.

Print	View
EL01	Event Logging 1 to 2 of 2
Action	Application ID
<input checked="" type="checkbox"/>	EDIFFS
	EDIMP

The Selection Criteria panel (EL02) is displayed. Some fields may already contain values.

EL01	Event Logging	1 to 2 of 2
Action	Application ID	
<input checked="" type="checkbox"/>	EDIFFS	
<div>EL02 Selection Criteria</div> <div>Application ID: EDIMP</div> <div>Date to</div> <div>Time to</div> <div>User ID to</div> <div>Format ID to</div> <div>Associated entry ID to</div> <div>Sort oldest first? N</div>		

3. Complete one or more of the fields as follows:

In this field:	Enter:
Date	A date or range of dates by which to select log entries. The format must match the date mask in the language profile you are using for this session.
Time	A time or range of times by which to select log entries. The format must match the time mask in the language profile you are using for this session.
User ID	A specific user ID, or a range of user IDs, by which to select log entries. For CICS, User ID is one of the following, in the order listed: <ol style="list-style-type: none"> 1. Sign-on user ID 2. Terminal ID 3. Application ID of the CICS region
Format ID	One of the following: <p>\$\$MSG-<i>msgid</i> where <i>msgid</i> is a six-character message identification used in DataInterchange messages. This format ID is used when messages are logged. When this data is displayed, breaks occur where blanks are found.</p> <p>\$\$MSG-xx where <i>xx</i> is the ID of the component logging the error message</p> <p>\$\$STD-<i>delimiters</i> where <i>delimiters</i> are the segment terminator, segment separator, data element delimiter, subelement delimiter, decimal notation, and release character. This format ID is used when standard data is logged. When this data is displayed, breaks occur where segment terminators are found.</p> <p>\$\$PSA-<i>profname</i> where <i>profname</i> is the name of the profile being accessed. This format ID is used whenever someone accesses a profile for which logging is active.</p> <p>\$\$DB2-(<i>errcode</i>) where <i>errcode</i> is the error code returned by DB2. This format ID is used whenever DB2 returns a serious error condition to DataInterchange. When this data is displayed, breaks occur at the beginning of each DB2 error message line, which starts with DSN.</p> <p>An application data format ID for entries logged as application data.</p> <p>Note: You can specify a partial Format ID to group your selection. For example, to select all messages logged by the translator, enter \$\$MSG-TR in the selection criteria field.</p>
Associated entry ID	A transaction handle or partial transaction handle.
Sort oldest first	Y to show the oldest log entry first, or N to show the most recent log entry first.

4. Press Enter.

If you are printing log entries, DataInterchange writes the entries to the print file you specified at logon, then redisplay the Event Logging panel (EL01). Use system facilities to print the file.

If you are viewing log entries, the View Log Entry panel (EL03) is displayed with the log entries that match the criteria you specified.

Event Logging

5. To make a specific line the first line of the display, type **L** and the line number on the command line, and press Enter.
6. To reverse the sequence of the sort, type **S** in the action column next to any item, and press Enter.
7. To view all of the log data for an entry, type **V** in the action column next to the entry you want to view, and press Enter.

Line Sort toggle View				
EL03		Event Log Entries		1 to 13 of 37
Application ID: EDIMP				
A	Line	Date	Time	Log data
-	0001	97/12/07	12:40:52	PS0302*08*A profile member with that name
-	0002	97/12/07	12:40:55	PS0302*08*A profile member with that name
-	0003	97/12/07	12:40:58	PS0302*08*A profile member with that name
-	0004	97/12/07	12:41:01	PS0302*08*A profile member with that name
-	0005	97/12/07	12:41:04	PS0302*08*A profile member with that name
-	0006	97/12/07	12:41:06	PS0302*08*A profile member with that name
-	0007	97/12/07	12:41:09	PS0302*08*A profile member with that name
-	0008	97/12/07	12:41:13	PS0302*08*A profile member with that name
V	0009	97/12/07	13:21:25	TR0403*04*Application data received will
-	0010	97/12/10	14:59:20	Profile log**Function = 94,Profile ID =
-	0011	97/12/10	15:00:37	Profile log**Function = 91,Profile ID =
-	0012	97/12/10	15:00:37	Profile log**Function = 10,Profile ID =
-	0013	97/12/10	15:01:05	Profile log**Function = 10,Profile ID =

The Log Data panel (EL04) is displayed.

EL04	Log Data	1 to 4 of 4
Application ID . . . :	EDIMP	Date : 97/12/14
User ID :	WORKERB	Time : 13:21:25
Format ID :	\$MSG-TR0403	
Associated entry ID :	E0000000000000000033	
TR0403*04*Application data received will be ignored. Structure TRAILER was not used in the Trading Partner Transaction Definition. Internal Trading Partner ID and Application Format = 12345 - POSEND. Transaction code, mode and function = 850 - ***PRODUCTION*** - SEND. ****		

“Interpreting Log Entries” on page 12-7 and “Translator Messages” on page 12-7 describe the data in a log entry.

8. When you are finished viewing the log entry, press F3 (Exit) twice to return to the Administrator’s Menu (MP01).

Interpreting Log Entries

Some of the data fields in a log entry are formatted so that you can find them easily. These include the date and time and several IDs, such as the ID of the log entry. Logged messages are unformatted and not as easy to read. The following key can help:

```

      1 1      (columns)
1    7 9 1 3
-----
Msg ID*nn*Text (variable length)**Symptoms (up to 256 bytes)

```

where:

* is used to separate the parts of the entry.

nn is a severity code:

00 Informational

04 Warning

08 Error

12 Severe error

For example:

```

TR1201*12*A program error occurred during anchor processing**
RC=12 ERC=01 FUNC=0212

```

The first two characters of the message ID identify the component detecting the error. In this example, TR indicates the translator. For more information, see *DataInterchange Messages and Codes*.

Translator Messages

The translator uses a unique format to report translation errors. It shows where and what the error is without having to log an image of the transaction. Figure 12-1 shows a sample translator message from the Audit Trail Report.

```

Message: TR0004 Severity: 04
Code in ID type field not found in validation table. Internal Trading Partner ID and Application Format =
TP112233 - POSEND. Transaction handle, code, mode, and function = 19900927115530000001 - 810 - PRODUCTION -
RECEIVE. Interchange, group, and transaction control numbers = 0000000009 - 26 - 0059. Current Loop-ID and
repetitions = 111000 - 2 - 3 - 1. Standard segment and field ID = NTE(005) - 8 - 3 - 2 - (2,0). Application
field ID = POLINEITEM - 10. Data type and value = AN - A1VALUE. Validation table name = CHKCODE.

```

Figure 12-1. Sample Translator Message

For more information about the Audit Trail Report, see the *DataInterchange Programmer's Reference*.

The following list describes the translator error messages.

- Internal trading partner ID and application format = TP112233 - POSEND
 - Internal trading partner ID is a name specified during mapping.
 - Application format is the ID of an existing data format.

Event Logging

- Transaction handle, code, mode, and function = 19900927115530000001 - 810 - PRODUCTION - RECEIVE

- Transaction handle is the ID from the Transaction Store.
- Code is the standard transaction or message code.
- Mode is PRODUCTION, TEST, or INFORMATION.
- Function is the current translator function.

- Interchange, group, and transaction control numbers = 000000009 - 26 - 0059

Control numbers from the interchange, group, and transaction or message service segments.

- Current loop-ID and repetitions = 111000 - 2 - 3 - 1

This example indicates that the error occurred in repetition 2 of loop 111000, within repetition 3 of its nesting loop, within repetition 1 of the next outer loop. You can have up to six repetition values.

- Standard segment and field ID = NTE(005) - 8 - 3 - 2 - (2,0)

The substitution values are: segment ID(sequence) - occurrence - repetition - field number - (data element, subelement)

This example indicates:

- The error occurred in the NTE segment defined in the standard as sequence number 005.
- This is the eighth segment within the received transaction, where the transaction header is segment number 1.
- This is the third repetition of the NTE(005) segment. This value is always 1 unless the segment is defined as a repeating segment.
- The error is in the second field within the NTE segment being received, where the segment ID is considered field number 1, and each data element or subelement contained in the received data increases the value by 1. The text (E05) will be used if the end of segment has been received. All subelements will be counted as present even if they are not physically present in the data.
- The error is in the second simple or composite data element, as defined in the standard, where the segment ID is considered to be field number 1.
- Because the second element is not a composite field, the subelement value is 0. If the error involved a subelement, the subelement value would indicate which component was in error, as defined in the standard, with the numbers starting at 1.

- Application field ID = POLINEITEM - 10

The substitution values are: structure name - offset

This example indicates that the error occurred in the field at offset 10 in structure POLINEITEM. (The first position is at offset 1.)

- Data type and value = AN - A1VALUE
- Validation table name = CHKCODE.

Turning Logging On and Off

You can control the logging of information at four points:

- Events associated with an application

The **Log flag** field in the activity log profile turns logging on and off for the application identified by the profile member. Turn logging on to record all events for the application. (Some types of event recording might be turned off even if this indicator is on.) Turn logging off to stop all event recording

for the application except for certain errors and events that change a transaction's status. Serious errors and changes in transaction status are always recorded.

- Events associated with a profile

The Log action on profile panels lets you control the logging of accesses to that profile. Logging is off by default. Turning logging on provides a record in the event log of each person who accessed the profile.

- Data in application format

The **Log application data** field in the transaction usage controls the logging of transaction data. Turn logging on if you want to log an image of the transaction. For outbound transactions, the image is made before translation; for inbound transactions, the image is made after the translator converts the data to your local format.

- Data in standard format

The **Log standard data** field in the trading partner profile controls the logging of standard data. Turn it on if you want to log an image of the standard segments. For outbound transactions, the image is made after the interchange trailer is completed. For inbound transactions, the image is made when a complete interchange envelope is received and before deenveloping.

Defining an Event Log

If your applications call DataInterchange directly, you can define a separate log for each application or one log that is shared by two or more applications. You can also define event logs for requesting services of the DataInterchange Utility. If your request includes the APPLID keyword, events are recorded in the log associated with that application ID.

To define a separate event log for an application, follow these steps:

1. Create an Application definitions profile (APPDEFS) with an ID that is up to eight characters long, unique, and easy to recognize.

For example, use PURCHASE for a purchasing application. If this log is for an application that calls DataInterchange, you must use this name when your application issues its first call statement to initialize DataInterchange. If you want the DataInterchange Utility to use the log associated with this application ID, use the APPLID keyword to pass the ID to the utility.

2. Add a member to the activity log profile (ACTLOGS). Use the application ID as the member name. The following is a sample member for an activity log profile:

```
Application ID . . . PURCHASE
Log file name . . . PURCHLOG
Log flag . . . . . 01
```

Figure 12-2. Sample Activity Log Profile Member

The Log file name is the name of the event log you want to associate with the application. Log flag is for activating event logging for the application. Set it to 01 to turn on logging. Set it to 00 to turn off logging. For a full description of this profile, see "Activity Log Profile" on page 12-1.

Event Logging

- | 3. If you are a VSAM user, add an allocate statement to the EDI CLIST provided by DataInterchange. The allocate statement is required for viewing the event log. For example:

```
ALLOC F(PURCHLOG) DS(EDI.PURCHASE.LOG) SHR
```

The allocate statement does not apply to CICS.

- | 4. If you are a VSAM user, create a VSAM file for the event log. Use the same definition that was used in the log file information profile. Give authorized users access to the file. See Appendix A, "Security," for details on giving access to files.

A DataInterchange event log can be used concurrently by multiple jobs. DataInterchange protects the integrity of the log by issuing MVS ENQueues and DEQueues to prohibit simultaneous updating by more than one process.

- | This concurrent updating of a VSAM file is allowed by VSAM when the SHARE OPTIONS for the file are defined as (3,3) or (4,3), which is the default specification for EDIMP and EDIFFS, the logs shipped with DataInterchange.

DataInterchange also supports the definition of an event log with SHARE OPTIONS of (2,3), which supports only a single updater at one time. If a log is defined with SHARE OPTIONS of (2,3), it is not necessary for DataInterchange to issue the MVS ENQueueus and DEQueues around each VSAM request. If significant logging is done and performance is a concern, a log with SHARE OPTIONS of (2,3) performs better than one with SHARE OPTIONS of (3,3). DataInterchange issues a single ENQueue for a SHARE OPTIONS (2,3) file when the log is opened, and a DEQueue when the file is closed, which effectively single-threads all processes using the log.

Chapter 13. Using the Interactive Entry Facility (IEF)

Before You Can Use IEF	13-3
Interactive Entry Facility Menu (MM01)	13-3
Working with Business Document Definitions	13-4
Adding a Business Document Definition	13-4
Updating, Viewing, or Deleting a Business Document Definition	13-6
Working with Business Documents	13-9
Business Documents Panel (DE01)	13-9
Titles of Business Document Panels	13-10
Business Document Status	13-10
Business Document Functions and Status	13-12
Navigating within a Business Document	13-13
Adding a Business Document	13-13
Sending a Business Document	13-17
Copying a Business Document	13-18
Deleting a Business Document	13-18
Filing, Archiving, or Getting (Restoring) a Business Document	13-19
Printing a Business Document	13-22
Updating a Business Document	13-22
Viewing a Business Document	13-23
Recalling a Business Document from the Network	13-24
Receiving a Business Document	13-25
Translating a Business Document to EDI Format	13-26
Working with Messages	13-26
Messages Panel (DM01)	13-26
Message Status	13-27
Message Functions and Status	13-28
Adding a Message	13-28
Sending a Message	13-29
Copying or Updating a Message	13-30
Deleting a Message	13-31
Printing a Message	13-31
Viewing a Message	13-31
Recalling a Message	13-31
Receiving a Message	13-32
Updating the Status of Messages	13-32
Handling File Requests	13-33
File Requests Panel (DF01)	13-33
File Request Status	13-34
File Request Functions and Status	13-34
Adding a Request to Send a File	13-35
Sending a File	13-36
Deleting a File Request	13-37
Viewing a File Request	13-37
Recalling a File	13-38
Receiving a File	13-38
Updating the Status of File Requests	13-39

Chapter 13. Using the Interactive Entry Facility (IEF)

The Interactive Entry Facility (IEF) lets you quickly convert from using paper documents to using electronic documents for exchanging with your trading partners. When you have a relatively small number of documents to send, IEF is a practical way for you to do business with your trading partners who require EDI. IEF also lets you work with documents without an application program, which means that you can become an EDI user quickly. IEF can also be used to test new applications.

IEF is designed for people who:

- Want a simple, easy-to-use introduction to EDI.
- Want to use EDI and still maintain a manual system.
- Need to electronically transmit a volume of documents too low to justify writing an application system.
- Need a tool to create model EDI documents to help develop applications for a high-volume business.

Once you have defined your business documents to IEF, you can:

- Add, copy, delete, update, and view a business document
- File, archive, and restore business documents
- Print business documents
- Translate and store the business document in standard format
- Send, receive, and recall business documents

| DataInterchange for MVS lets you send, receive, and recall files. You can also add, delete, and view
| requests to send files. This function is not available in DataInterchange for CICS.

IEF also lets you add, copy, delete, update, view, send, receive, recall, and print free-form messages such as schedule changes, plant closings, and short memorandums.

Figure 13-1 on page 13-2 shows the IEF components and how they are connected. Notice that business documents are central to what you work with when you use IEF.

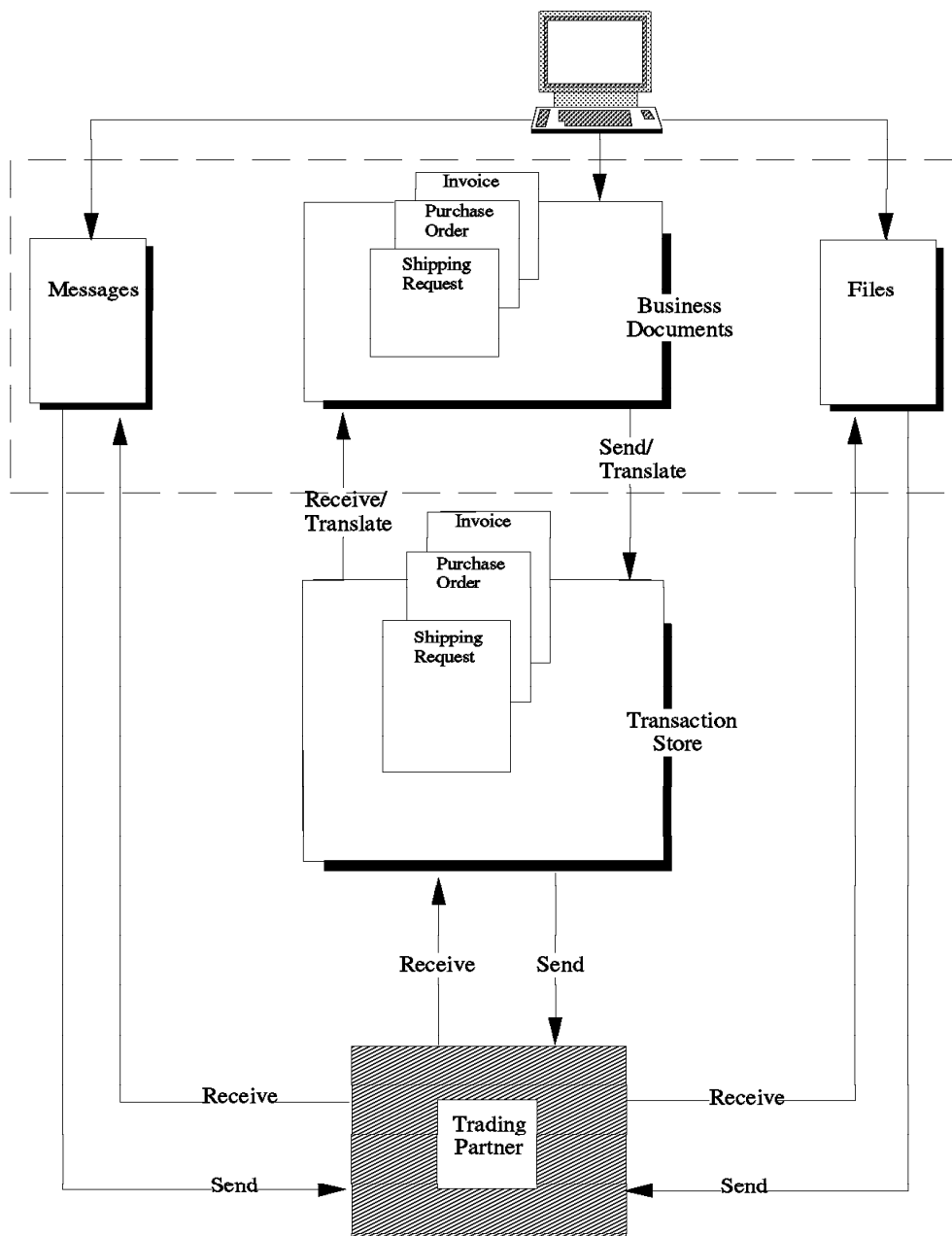


Figure 13-1. IEF Components

You can use the same IEF document type for all your trading partners, even if each trading partner has different EDI requirements. For example, you can enter invoices the same way for different trading

partners. IEF then translates the invoices to meet the specific requirements of each trading partner before it sends them.

DataInterchange uses file requests to keep track of file sends, receives, and recalls. However, the file requests function is not available for DataInterchange for CICS.

Before You Can Use IEF

Before you can use IEF, you need the following information:

- Application data format ID
- Expected values and data types for the various fields
- Internal trading partner ID of the trading partner you will be doing business with
- Trading partner nickname
- Your requestor ID
- Specific names for transmission functions
- Names of files for filing or archiving business documents

Interactive Entry Facility Menu (MM01)

To use the Interactive Entry Facility Menu (MM01), follow these steps:

1. From the Administrator's Menu (MP01), select Interactive Entry Facility. The Interactive Entry Facility Menu (MM01) is displayed.

```
MM01                      Interactive Entry Facility

Type the number of your selection below and press Enter, or press the Exit
key to exit.

Choice ==>  _  1. Business documents
               2. Messages
               3. File requests
               4. Business document definitions
```

This menu lists the Interactive Entry Facility tasks that you are authorized to select.

2. Select a task by typing its number or moving the cursor to your choice, and press Enter. Table 13-1 on page 13-4 describes the Interactive Entry Facility Menu (MM01) options.

Table 13-1. Interactive Entry Facility Menu (MM01) Options

Choice	Description
Business documents	Work with a business document, such as a purchase order or an invoice. You can add, copy, delete, file, print, update, view, recall, receive, send, or translate business documents.
Messages	Work with free-form messages. You can send messages to a trading partner, recall messages from a network, or receive messages from a trading partner. You can add, delete, print, update, copy, or view messages. You can also request an update of message statuses.
File requests	Work with file requests. You can add, delete, or view requests to send a file. You can send or receive a file, or recall a file from the network. You can also update the status of file requests.
	This option is not available in DataInterchange for CICS.
Business document definitions	Create and maintain the business document definitions which are used by IEF. You can add, delete, update, and view business document definitions.

Working with Business Document Definitions

The Interactive Entry Facility uses the business document definition, which is based on an application data format, to properly process a business document. The business document definition contains the defaults for sending business documents and the default file names for storing and archiving business documents.

You can add, delete, update, and view business document definitions.

Adding a Business Document Definition

To add a business document definition, follow these steps:

1. From the Interactive Entry Facility Menu (MM01), select Business document definitions. The Business Document Definitions panel (DD01) is displayed.
2. Type **A** in the action column next to any item, and press Enter.

Add	Delete	Update	View
DD01		Business Document Definitions	1 to 4 of 4
A		Data Format ID	Description
a		GENINV	General Use Invoice - Send
-		GENINRCV	General Use Invoice - Receive
-		GENPO	General Use Purchase Order - Send
-		GENPORCV	General Use Purchase Order - Receive

Note: If there are no existing document definitions, the Add Business Document Definition panel (DD02), shown in the next step, is displayed.

The Add Business Document Definition panel (DD02) is displayed.

```

DD02                      Add Business Document Definition

Data format ID  . .  posend _____

File to . . . . . _____
Archive to . . . . . _____
File formatted to . . . . . _____

Send defaults

Requestor ID  . . . . . reqid _____ +

Specific name
  Use default? (Y/N)  . .  y +
  Display panel? (Y/N) . .  y +

Network Acknowledgments
  Use defaults? (Y/N)  . .  y +
  Display panel? (Y/N) . .  y +

```

3. Complete the fields as follows:

In this field:

Data format ID

Type:

The ID of the existing application data format you want to use for this document definition.

File to

The name of the default file in which documents associated with this document definition are placed when filed as unformatted data. If you do not provide a file name here, you will have to provide one in the **File to** field of the File business document panel (DE09) when you file a document.

DataInterchange for CICS displays two fields for you to specify the file name and file type. The file type can be a temporary storage queue (TS), transient data queue (TD), or a VSAM entry sequenced data set (VS).

Archive to

The name of the default file in which documents associated with this document definition are placed when archived. If you do not provide a file name here, you will have to provide one in the **Archive to** field of the Archive business document panel (DE08) when you archive a document.

DataInterchange for CICS displays two fields for you to specify the file name and file type. The file type can be a temporary storage queue (TS), transaction data queue (TD), or a VSAM entry sequenced data set (VS).

File formatted to

The name of the file in which documents associated with this document definition are placed when filed as formatted text. If you do not provide a file name here, you will have to provide one in the **File to** field of the File Formatted business document panel (DE13) when you file a formatted document.

DataInterchange for CICS displays two fields for you to specify the file name and file type. The file type can be a temporary storage queue (TS), transient data queue (TD), or a VSAM entry sequenced data set (VS).

Requestor ID

The requestor ID to be used by IEF when you use the default for identifying the sender or receiver of business documents.

In this field:
Specific name

Type:
For Use default, type **Y** if you want to use the message user class specified in the requestor profile as the default specific name on the Network Access panel (DE14). Type **N** if you do not want the name in the profile used as a default. Specific name and message user class are equivalent.

For Display panel, type **Y** if you want the Network Access panel (DE14) displayed, giving you the chance to modify the defaults. Type **N** if you want to bypass the Network Access panel (DE14) and always use the default values.

You cannot specify **N** for both the Use defaults and Display panel fields. If you do not use the defaults, you must display the panel. If you do not display the panel, you must use the defaults.

If your network does not support receiving documents by a specific name, the **Specific name** field on the Network Access panel (DE14) is not displayed.

Network
Acknowledgments

For Use defaults, type **Y** if you want to use the network acknowledgment value specified in the trading partner profile as the default on the Network Acknowledgments panel (DE11). Type **N** if you do not want to use the value in the profile as a default.

For Display panel type **Y** if you want the Network Acknowledgments panel (DE11) displayed, giving you the chance to modify defaults. Type **N** if you want to bypass the Network Acknowledgments panel (DE11) and always use the default values.

You cannot specify **N** for both the Use defaults and Display panel fields. If you do not use the defaults, you must display the panel. If you do not display the panel, you must use the defaults.

The network you and your trading partner use determines what appears on the Network Acknowledgments panel (DE11). Only those acknowledgments that your network supports are shown on this panel. If your network does not support any acknowledgments, this panel is not displayed.

4. Press Enter to save this information.

The Business Documents panel (DE01) is redisplayed with the new business document definition added to the list.

5. Press F3 (Exit) to return to the Interactive Entry Facility Menu (MM01).

Updating, Viewing, or Deleting a Business Document Definition

To update, view, or delete a business document definition, follow these steps:

1. From the Interactive Entry Facility Menu (MM01), select Business document definitions. The Business Document Definitions panel (DD01) is displayed.

2. To delete a business document definition:

a. Type **D** in the action column next to the item you want to delete, and press Enter.

Add Delete Update View		
<hr/>		
DD01	Business Document Definitions	1 to 5 of 5
A	Data Format ID	Description
-	GENINV	General Use Invoice - Send
-	GENINRCV	General Use Invoice - Receive
-	GENPO	General Use Purchase Order - Send
-	GENPORCV	General Use Purchase Order - Receive
d	POSEND	Purchase Order - Send

The Delete Business Document Definition panel (DD03) is displayed.

<hr/>		
DD01	Business Document Definitions	1 to 5 of 5
A	Data Format ID	Description
-	GENINV	
-	GENINRCV	
-	GENPO	
-	GENPORCV	
d	POSEND	

DD03 Delete Business Document Definition

Data format ID . . : POSEND
Description . . . : Purchase Order - Send

Press Enter to delete this definition. Press the
Cancel key to cancel the request.

DataInterchange will not delete a business document definition if a business document exists for the definition. You will see a message if you try to delete a definition with an existing business document.

b. Press Enter to confirm the deletion. The Business Document Definitions panel (DD01) is redisplayed.

3. To update a business document definition:

- a. Type **U** in the action column next to the item you want to update, and press Enter. The Update Business Document Definition panel (DD04) is displayed.

DD04

Update Business Document Definition

Data format ID : POSEND

Data format description . : Purchase Order - Send

File to : _____

Archive to : _____

File formatted to . : _____

Send defaults

Requestor ID : REQID_____ +

Specific name

Use default? (Y/N) . . : Y+

Display panel? (Y/N) . . : Y+

Network Acknowledgments

Use defaults? (Y/N) . . : Y+

Display panel? (Y/N) . . : Y+

You can update any field except the **Data format ID** and **Data format description** fields.

- b. When you are finished updating the business document definition, press Enter to return to the Business Document Definitions panel (DD01).

4. To view a business document definition:

- a. Type **V** in the action column next to the item you want to view, and press Enter. The View Business Document Definition panel (DD05) is displayed.

DD05

View Business Document Definition

Data format ID : POSEND

Data format description . : Purchase Order - Send

File to :

Archive to :

File formatted to :

Send defaults

Requestor ID : REQID

Specific name

Use default? (Y/N) . . : Y

Display panel? (Y/N) . . : Y

Network Acknowledgments

Use defaults? (Y/N) . . : Y

Display panel? (Y/N) . . : Y

- b. When you are finished viewing the business document definition, press Enter to return to the Business Document Definitions panel (DE01).

5. Press F3 (Exit) to return to the Interactive Entry Facility Menu (MM01).

Working with Business Documents

A business document is a collection of data associated with an application data format and a business document definition.

When working with business documents, you can:

- Add a business document
- Send a business document to a trading partner
- Copy a business document
- Delete a business document
- File, archive, or restore a business document
- Print a business document
- Update a business document
- View a business document
- Recall a business document from the network
- Receive a business document from the network
- Translate a business document to EDI format

Business Documents Panel (DE01)

When you work with business documents, you select actions from the Business Documents panel (DE01). To access this panel, follow these steps:

1. From the Interactive Entry Facility Menu (MM01), select Business documents. If there are no existing business documents, the Select Business Document Action panel (DE22) is displayed.

```
DE22                      Select Business Document Action

No documents exist that you are authorized to use. Type the number of
your selection below and press Enter, or press the Exit key to exit.

Choice==>  _  1. Add business document
              2. Receive business document
              3. Get business document
```

2. To add a business document, select **Add business document**, then see step 2 under “Adding a Business Document” on page 13-13.
3. To receive a business document, select **Receive business document** then see step 2 under “Receiving a Business Document” on page 13-25.
4. To get an archived business document, select **Get business document**, then see step 5a under “Filing, Archiving, or Getting (Restoring) a Business Document” on page 13-19.

If there are existing business documents, the Business Documents panel (DE01) is displayed.

Add Copy Delete File Print Update View recaLl Receive Send Translate						
DE01		Business Documents				1 to 4 of 4
A	Document Key	Trading Partner	Data Format ID	Status	Date	
-	P0123456	PISCES	GENPO	SEND REQUESTED	97/12/15	
-	P0123456	CAPRICOR	GENPORCV	RECEIVE TRANSLATED	97/12/15	
-	P0567890	ARIES	GENINV	SEND TRANSLATED	97/12/10	
-	13579	LEO	GENINRCV	RECEIVE TRANSLATED	97/12/03	

The Business Documents panel (DE01) contains the following information:

Column	Description
Document Key	The name of the document, which is created from data in the application control fields defined by the application data format.
Trading Partner	The nickname of the trading partner associated with this document.
Data Format ID	The ID of the application data format associated with the document.
Status	The status of the document. See “Business Document Status” for details.
Date	For an outbound business document that has not been sent, the date of creation or the last update. For an outbound business document that has been sent, the date the document was sent. For an inbound business document, the date the document was received and translated.

Titles of Business Document Panels

The titles on the business document panels are the function, such as **Add**, followed by the description of the associated application data format. For example, if you are adding a business document that is associated with an application data format described as Purchase Order - Send, the title on the Add business document panel (DE03) would be:

Add Purchase Order - Send

Business Document Status

The following table describes the status values for business documents. Some of these status values are similar to the transaction status codes described in “Transaction Status Codes” on page 10-28, but these values apply only to business documents within IEF.

Table 13-2 (Page 1 of 2). Business Document Status

Status	Description
Incomplete	Document is not ready to be sent.
Ready	Document is ready to be sent, but the person who has prepared it does not have the authority to send it.
Held	The document is complete and ready to be sent, but is being held for sending at a later time.
Status pending	The Transaction Store has not received the document’s status yet.
Send requested	DataInterchange successfully issued a send command to the network. Document is translated and an immediate send has been requested.

Table 13-2 (Page 2 of 2). Business Document Status

Status	Description
Sent to network	DataInterchange successfully sent a message to the network.
Recall requested	DataInterchange successfully issued a recall command.
Send translated	Document was translated successfully and stored in the Transaction Store.
Send translate error	A translation error occurred. However, the document was stored in the Transaction Store successfully.
Enveloped	The transaction is enveloped and ready for sending. The document was translated successfully, but could not be sent.
Envelope error	An error occurred during enveloping. The document is still in the Transaction Store.
Send request error	DataInterchange encountered an error while attempting to issue a send command to the network.
Not sent-network error	The network encountered errors that prevented the document from being sent.
Sent with errors	The network completed the send with minor errors.
Accepted by network	DataInterchange received a network receipt acknowledgment for this business document.
Delivered by network	DataInterchange received a network delivery acknowledgment for this business document.
Purged by network	DataInterchange received a network purge acknowledgment for this business document.
Received	The document was successfully received from the network and deenveloped.
Transaction accepted with errors	DataInterchange received a functional acknowledgment indicating that the document was acceptable, but that errors were noted.
Recall request error	DataInterchange encountered an error while attempting to issue a recall command to the network.
Recalled	DataInterchange received a network cancel acknowledgment for this business document.
Transaction accepted	DataInterchange received a functional acknowledgment that the data is good or acceptable to the receiver.
Transaction rejected	DataInterchange received a functional acknowledgment indicating that the data is in error.
Trx detached-send	The envelope for this transaction was replaced by another envelope with an identical key. This transaction is not included in the new envelope. In effect, the transaction is detached from its envelope. This can occur if you reset the control numbers in the trading partner profile. If you still want to use this transaction, you can reenvelope or reenvelope and send it using the Transaction Store facility or DataInterchange Utility.
Receive translated	The document was translated successfully. If there were translation errors, the error level is equal to, or less than, the level specified in the trading partner usage for the transaction mapping.
Receive translate error	A translation error occurred. The error level is higher than the error level specified as acceptable in the trading partner usage for the transaction mapping. The standard data remains in the Transaction Store.
Trx detached-recv	You received an envelope with the same key as the envelope that contained this transaction. This transaction is not included in the new envelope. In effect, the transaction is detached from its envelope. This can occur when your trading partner sends the same envelope more than once.

Business Document Functions and Status

A document's status determines which functions are valid. The following table shows which functions are valid for each status value, where X indicates that the function is valid. For example, if a document's status is SEND REQUESTED, you can copy, delete, file, print, or view the document, but you cannot update, recall, send, or translate it.

Table 13-3. Business Document Functions and Status

Status	Copy	Delete	File	Print	Update	View	Recall	Send	Translate
Incomplete	X	X	X	X	X	X			
Ready	X	X	X	X	X	X		X	X
Held	X	X	X	X	X	X		X	X
Send requested	X	X	X	X		X	X		
Sent to network	X	X	X	X		X	X		
Recall requested	X	X	X	X		X			
Send translated	X	X	X	X		X		X.	
Send translate error	X	X	X	X		X		X	X
Send request error	X	X	X	X	X	X		X	X
Not sent net error	X	X	X	X	X	X		X	X
Sent with errors	X	X	X	X		X	X		
Accepted by network	X	X	X	X		X	X		
Delivered by network	X	X	X	X		X			
Purged by network	X	X	X	X	X	X		X	X
Recalled	X	X	X	X	X	X		X	X
Recalled request error	X	X	X	X		X	X		
Trans rejected	X	X	X	X		X			
Trans accepted	X	X	X	X		X			
Trx detached - send	X	X	X	X		X	X		
Enveloped	X	X	X	X		X			
Envelope error	X	X	X	X		X			
Received trans error		X	X	X		X			
Receive translated		X	X	X		X			
Trx detached - recv		X	X	X		X			

Navigating within a Business Document

The following list describes how to navigate within a business document on the Add (DE03), Copy (DE04), Update (DE06), and View (DE07) business document panels:

- To display the fields and structures of a structure, type an occurrence number in the entry field of that structure, and press Enter.
- To display the previous occurrence of a structure, press Previous (F7). This command is not available for the first occurrence of a structure.
- To display the next occurrence of a structure, press Next (F8). This command is not available for the last occurrence of a structure.
- To validate the data you typed on a panel, press Enter. If errors exist, they are highlighted. If you are typing data for occurrences, pressing Enter validates the data, then proceeds to the next occurrence.

Validation is based on the data type specified in the application data format, and any edits and translation or validation tables specified in the trading partner transaction mapping.

Table 7-2 on page 7-17 describes the data types for application data formats; Chapter 9, "Mapping Your Application Data to an EDI Standard Transaction Set" describes trading partner transaction mappings; and Chapter 8, "Translation and Validation Tables" describes the translation and validation tables.

- To cancel any changes you have made to occurrences, press Cancel (F12). Press Cancel (F12) again after the warning message is displayed.
- To return to the base structure, press F3 (Exit).
- To leave the business document, return to the base structure, if necessary, and press F3 (Exit).

Adding a Business Document

To add a business document, follow these steps:

1. On the Business Documents panel (DE01), type **A** in the action column next to any item, and press Enter. The Select Data Format panel (DE02) is displayed, listing the data format defined in a business document definition.
2. In the **Choice** field, type the number of the application data format you want to use for this business document, and press Enter.

DE02	Select Data Format	1 to 5 of 5
Type the number of the data format for the document you wish to add.		
Choice ==> ____	Data Format ID	Description
	1. GENINV	General Use Invoice - Send
	2. GENINRCV	General Use Invoice - Receive
	3. GENPO	General Use Purchase Order - Send
	4. GENPORCV	General Use Purchase Order - Receive
	5. POSEND	Purchase Order - Send

The Trading Partner Selection panel (DE23) is displayed.

DE02		Select Data Format	1 to 5 of 5
Type the number of the data format for the document you wish to add.			
Choice ==>	__5	1. GENINV	General Invoice - Send
<div> <div>DE23</div> <div>Trading Partner Selection</div> <div> Trading Partner _____ Usage indicator (P/T/I) _ + </div> </div>			

3. Complete the fields as follows:

- In the **Trading Partner** field, type the nickname of the trading partner you want to use with this business document.
- In the **Usage indicator (P/T/I)** field, enter a P to use a production usage. Enter a T to use a test usage. Enter an I to use an information usage.
- Press Enter.

If this trading partner has more than one internal trading partner ID, the Internal Trading Partner ID Selection panel (DE17) is displayed.

- In the **Choice** field, type the number of the internal trading partner ID you want to use for this document, and press Enter.

DE02		Select Data Format	1 to 5 of 5
Type the number of the data format for the document you wish to add.			
Choice ==>	__5	1. GENINV	General Invoice - Send
<div> <div>DE23</div> <div>Trading Partner Selection</div> <div> <div>DE17</div> <div>Internal Trading Partner ID Selection</div> <div>Select the internal trading partner ID you wish for this document.</div> <div> Choice ==> ____ <div> 1. ACCTING 2. WAREHOUSE </div> </div> </div> </div>			

The Enter Document Key panel (DE19) is displayed over the Add business document panel (DE03).

DE03		Add Purchase Order - Send	1 to 5 of 5
Data format ID . . .	POSEND		
Document key . . .	_____	Update? (Y/N)	Y
Trading partner . . .	PISCES	Test transaction :	N
<div> <div>DE19</div> <div>Enter Document Key</div> <div>1 to 1 of 1</div> <div>Fill in the data for each field.</div> <div>PONUMBER . . _____</div> </div>			

The Enter Document Key panel (DE19) lists the fields that define the application control number in the application data format. In this example, the application control number consists only of the purchase order number.

5. Complete the fields on this panel, and press Enter.

The Add business document panel (DE03) is displayed. This panel shows the base structure of the application data format.

DE03		Add Purchase Order - Send	1 to 5 of 5
Data format ID . . :	POSEND		
Document key . . . :	654321	Update? (Y/N)	N
Trading partner . . :	PISCES		Test transaction : N
Structure :	POBASE	Occurrence 1 of 1	
<p>Type data in the fields provided and/or select structures by typing the number of the first occurrence you wish to work with.</p>			
	PODATE		
	PONUMBER :	654321	
___	LINEITEMS	Occurs	5 times
	TOTALITEMS . . .		
	COMMENT		

A base structure describes a document's boundaries. The base structure contains all of the data fields. These data fields can be grouped into subordinate structures, such as the fields making up a company name and address, or the fields making up a line item. Some structures repeat within a document, while others do not. For example, a name and address structure may occur only once in a business document, while the line item may occur several times.

Fields have input areas after them, such as the following:

PODATE _____

Structures have input areas before them, such as the following:

___ LINEITEMS Occurs 5 times

6. Complete the data fields as appropriate.

7. If you need to update the values in the fields that define the document key, type **Y** in the **Update?** field located in the top right corner of the panel, and press Enter.

The Enter Document Key panel (DE19) is displayed with the existing values.

8. Make your changes, and press Enter.

9. To add the data for the structures, type an occurrence number in the field in front of the structure, and press Enter.

The Add business document panel (DE03) now shows the fields and structures of the structure you selected in the previous step.

DE03	Add Purchase Order - Send	1 to 5 of 5
Data format ID . . .	POSEND	
Document key . . .	654321	
Trading partner . . .	PISCES	Test transaction : N
Structure	LINEITEMS	Occurrence 1 of 5
within POBASE(1)		
Type data in the fields provided.		
PARTNUMBER . . .	_____	
DESCRIPTION . . .	_____	
QUANTITY	_____	
UNITOFMEASURE	_____	
UNITPRICE	_____	

10. Complete the fields as appropriate, and press Enter to validate and save the data you typed. DataInterchange redisplay this panel for the next occurrence. See “Navigating within a Business Document” on page 13-13 for more information on moving through the business document.
11. When you are finished entering data for this business document, return to the base structure, if necessary, and press F3 (Exit) to display the Status Choices Menu (DE10).

DE03	Add Purchase Order - Send	1 to 5 of 5
Data format ID . . .	POSEND _____	
Document key . . .	654321 _____	Update? (Y/N) N
Trading partner . . .	PISCES _____	Test transaction : N
Structure		
Type data in the f the number of the		
PODATE . . .		
PONUMBER . . .		
— LINEITEMS		
TOTALITEMS .		
COMMENT . . .		

DE10	Status Choices
Select the status you wish for this document.	
Choice ==> _	1. Send now 2. Translate and store 3. Hold for send at a later time 4. Incomplete - not ready for send 5. Cancel this new document

12. To translate and send the document now, select **Send now**, then go to step 2 under “Sending a Business Document” on page 13-17.

Otherwise, select one of the other options:

Select:

Translate and store

Hold for send at a later time

Incomplete – not ready for send

Cancel this new document

To:

Translate and store the document in the Transaction Store

Hold the completed document for sending later

Save the document to work on later

Discard this new document

DataInterchange processes the business document according to the option you selected, then redisplay the Business Documents panel (DE01).

Sending a Business Document

To send a business document from the Business Documents panel (DE01), follow these steps:

- 1. Type **S** in the action column next to the business document you want to send, and press Enter. The Network Acknowledgments Send panel (DE11) is displayed if you specified in the business document definition (page 13-6) that you wanted to see this panel.

Only the acknowledgments that your network supports are shown on this panel. If your network does not support any acknowledgments, this panel is not displayed.

If you specified the use of defaults in the business document definition (page 13-6), some of the acknowledgments may already be selected.

- 2. Type a slash (/) next to each type of acknowledgment you want to receive, and press Enter.

DE01

Business Documents

1 to 7 of 7

	Document Key	Trading Partner	Data Format ID	Status	Date
A	P0123456				
-	P0123456				
-	P01234568906				
-	P0654321				
S	072367				
-	13579				
-	654321				

DE11

Network Acknowledgments: Send

Place a '/' next to the acknowledgments you wish to receive and press Enter, or just press Enter if you wish to receive no acknowledgments.

☒ Accepted by network

☐ Delivery to trading partner

☐ Purged by network

The Network Access panel (DE14) is displayed if you specified in the business document definition (page 13-6) that you wanted to see this panel.

DE01

Business Documents

1 to 7 of 7

	Document Key	Trading Partner	Data Format ID	Status	Date
A	P0123456				
-	P0123456				
-	P01234568906				
-	P0654321				
S	072367				
-	13579				
-	654321				

DE11

Network Acknowledgments: Send

DE14

Network Access

Requestor ID

Specific name

☐ Purged by network

If your network does not support receiving documents by a specific name, the **Specific name** field is not displayed.

If you specified the use of defaults in the business document definition (page 13-6), these fields may already contain values.

3. Complete the fields as follows:

- In the **Requestor ID** field, type your requestor ID.
- In the **Specific name** field, type the name for a specific type of data. For example, you could use a department name, such as DEPT03, or the type of document, such as PURCHORD for purchase order. The receiver can use the value in this field for selective data retrieval.
- Press Enter.

DataInterchange translates and sends the business document, then redisplay the Business Documents panel (DE01) with the status of the business document updated.

Copying a Business Document

To copy a business document, follow these steps:

- On the Business Documents panel (DE01), type **C** in the action column next to the business document you want to copy, and press Enter.
- Continue with step 2 under “Adding a Business Document” on page 13-13. The only difference is that the Copy business document panel (DE04) is used rather than the Add business document panel (DE03).

Deleting a Business Document

To delete a business document, follow these steps:

- On the Business Documents panel (DE01), type **D** in the action column next to the business document you want to delete, and press Enter.
The Delete business document panel (DE05) is displayed.
- Press Enter to delete this business document.

DE01		Business Documents			1 to 2 of 2	
A	Document Key	Trading Partner	Data Format ID	Date	Status	
	123456					
d	654321					

DE05 Delete Purchase Order - Send

Data format ID . . . : POSEND
Trading partner . . . : PISCES
Document key . . . : 654321

Press Enter to delete the document. Press the Cancel key to cancel the request.

DataInterchange deletes the business document, then redisplay the Business Documents panel (DE01).

Filing, Archiving, or Getting (Restoring) a Business Document

Using the File Options Menu (DE15), you can file unformatted or formatted documents, archive documents, or get documents from an archive.

A filed document is copied to a storage file, but remains in the list on the Business Documents panel (DE01). When you want to store a readable presentation of document data, file it as formatted data. Unformatted and formatted business documents must be stored separately.

An archived document is added to an archive file, then removed from the list on the Business Documents panel (DE01). Archive a document if you want to retain a copy of the data in application format, but you do not want that copy in IEF.

Unformatted and archived documents must have a variable block record format and a block size of 32754 bytes. Formatted documents must have a fixed block record format and a logical record length of 132 characters.

You can provide default file names for unformatted data, formatted data, and archived data in the business document definition. See “Adding a Business Document Definition” on page 13-4 for details.

To file a business document, follow these steps:

1. On the Business Documents panel (DE01), type **F** in the action column next to the business document you want to file, and press Enter. The File Options Menu (DE15) is displayed.

DE15	File Options
Select the type of file function.	
Choice ==> _	<ol style="list-style-type: none">1. File data - move the document data to a file2. Archive data - move the document data to a file and delete the document from the list3. File formatted - format the document data and move the formatted data to a file4. Get data - restore the document data from an archive file and add the document to the list

2. To file the business document as is:

- a. Type **1** in the selection field, and press Enter. The File business document panel (DE09) is displayed.

DE15	File Options										
Select the type of file function.											
Choice ==> 1	1. File data - move the document data to a file										
<table border="1"><thead><tr><th>DE09</th><th>File Purchase Order - Send</th></tr></thead><tbody><tr><td>Data format ID . .</td><td>POSEND</td></tr><tr><td>Trading partner . .</td><td>PISCES</td></tr><tr><td>Document key . . .</td><td>654321</td></tr><tr><td>File to</td><td>_____</td></tr></tbody></table>		DE09	File Purchase Order - Send	Data format ID . .	POSEND	Trading partner . .	PISCES	Document key . . .	654321	File to	_____
DE09	File Purchase Order - Send										
Data format ID . .	POSEND										
Trading partner . .	PISCES										
Document key . . .	654321										
File to	_____										

b. Type a file name, if necessary, and press Enter.

DataInterchange appends the business document data to the file, then redisplay the Business Documents panel (DE01).

- a. Type **2** in the selection field of the File Options panel (DE15), and press Enter. The Archive business document panel (DE08) is displayed.

The **Archive to** field contains the default file name, if you specified one in the business document definition.

Note: Using a different file name here does not change the default in the business document definition.

DataInterchange appends the business document data to the file, removes the business document from the business documents database, then redisplay the Business Documents panel (DE01).

4. To file the business document with formatted data:

- a. Type **3** in the selection field of the File Options panel (DE15), and press Enter. The File formatted business document panel (DE13) is displayed.

DE15		File Options																					
Select the type of file function.																							
Choice ==>	3	<ol style="list-style-type: none">1. File data - move the document data to a file2. Archive data - move the document data to a file and delete the document from the list3. File formatted - format the document data and move the formatted data to a file																					
<div><table border="1"><thead><tr><th colspan="2">DE13</th><th colspan="2">File formatted Purchase Order - Send</th></tr></thead><tbody><tr><td>Data format ID . .</td><td>:</td><td colspan="2">POSEND</td></tr><tr><td>Trading partner . .</td><td>:</td><td colspan="2">PISCES</td></tr><tr><td>Document key . . .</td><td>:</td><td colspan="2">654321</td></tr><tr><td>File to</td><td>:</td><td colspan="2"></td></tr></tbody></table></div>				DE13		File formatted Purchase Order - Send		Data format ID . .	:	POSEND		Trading partner . .	:	PISCES		Document key . . .	:	654321		File to	:		
DE13		File formatted Purchase Order - Send																					
Data format ID . .	:	POSEND																					
Trading partner . .	:	PISCES																					
Document key . . .	:	654321																					
File to	:																						

The **File to** field contains the default file name, if you specified one in the business document definition.

- b. Type a file name, if necessary, and press Enter.

Note: Using a different file name here does not change the default in the business document definition.

DataInterchange appends the business document data to the file, then redisplay the Business Documents panel (DE01).

5. To get an archived business document:

- a. Type **4** in the selection field of the File Options panel (DE15), and press Enter. The Specify File to Restore panel (DE20) is displayed.

DE15		File Options									
Select the type of file function.											
Choice ==>	4	<ol style="list-style-type: none">1. File data - move the document data to a file2. Archive data - move the document data to a file and delete the document from the list3. File formatted - format the document data and move the formatted data to a file4. Get data - restore the document data from an archive file and add the document to the list									
<div><table border="1"><thead><tr><th colspan="2">DE20</th><th colspan="2">Specify File to Restore</th></tr></thead><tbody><tr><td>Archive file name . .</td><td>:</td><td colspan="2"></td></tr></tbody></table></div>				DE20		Specify File to Restore		Archive file name . .	:		
DE20		Specify File to Restore									
Archive file name . .	:										

The **Archive file name** field contains the default file name, if you specified one in the business document definition.

- b. Type a file name, if necessary, and press Enter.

Note: Using a different file name here does not change the default in the business document definition.

The Archived Documents panel (DE21) is displayed, listing the business documents in this file.

- c. Type **R** in the action column next to the business documents you want to restore.

Restore					
DE21		Archived Documents			1 to 2 of 2
A	Document Key	Trading Partner	Data Format ID	Status	Archive Date
—	072367	PISCES	GENPO	SEND REQUESTED	97/12/01
—	072569	PISCES	GENPO	RECEIVED	97/12/02

DataInterchange retrieves the archived business documents, and adds them to the business documents database.

- d. Press F3 (Exit) to return to the Business Documents panel (DE01).

Printing a Business Document

To print a business document, on the Business Documents panel (DE01), type **P** in the action column next to the business document you want to print, and press Enter.

DataInterchange sends the business document data to the printer, print queue, or print file that you specified at logon.

Updating a Business Document

To update a business document, follow these steps:

1. On the Business Documents panel (DE01), type **U** in the action column next to the business document you want to update, and press Enter. The Update business document panel (DE06) is displayed.
2. Make the changes on this panel and for the structures. See “Navigating within a Business Document” on page 13-13 for information on moving through the business document.

DE06	Update Purchase Order - Send	1 to 5 of 5
Data format ID . . . : POSEND		
Document key. . . : 654321		
Trading partner . . . : PISCES		
Structure : LINEITEMS		
		Test transaction : N
		Occurrence 1 of 1
Type data in the fields provided and/or select structures by typing the number of the first occurrence you wish to work with.		
PODATE 971214		
PONUMBER : 654321		
LINEITEMS Occurs 5 times		
TOTALITEMS		
COMMENT		

- When you are finished updating the business document, return to the base structure, and press F3 (Exit) to display the Status Choices Menu (DE10).

DE06	Update Purchase Order - Send	1 to 5 of 5
Data format ID . . :	POSEND	
Document key. . . :	654321	
Trading partner . . :	PISCES	Test transaction : N
Structure :		
Type data in the file		
the number of the file		
PODATE . . .		
PONUMBER . .		
LINEITEMS		
TOTALITEMS .		
COMMENT . .		

DE10 Status Choices

Select the status you wish for this document.

Choice ==> 1. Send now

 2. Translate and store

 3. Hold for send at a later time

 4. Incomplete - not ready for send

 5. Cancel this update

- To translate and send the document now, select **Send now**, then go to step 2 under “Sending a Business Document” on page 13-17.

Otherwise, select one of the other options:

Select:

Translate and store

Hold for send at a later time

Incomplete – not ready for send

Cancel this update

To:

Translate and store the document in the Transaction Store

Hold the document for sending later

Save the document to work on later

Discard the changes made in this update

DataInterchange processes the business document according to the option you selected, then redisplays the Business Documents panel (DE01).

Viewing a Business Document

To view a business document, follow these steps:

- On the Business Documents panel (DE01), type **V** in the action column next to the business document you want to view, and press Enter.

If a customized document layout exists for this document, the View Customized Business Document panel (DE18) is displayed.

DE18

View Customized Document

More: +

XYZ Company

Some address

Purchase order number 654321

Supplier

Purchase order date 971214

Invoice to

Supplier

Ship to

Otherwise, the View business document panel (DE07) is displayed.

DE07

View Purchase Order - Send

1 to 5 of 5

Data format ID . . . : POSEND

Document key. . . . : 654321

Trading partner . . . : PISCES

Structure : LINEITEMS

Test transaction : N

Occurrence 1 of 1

Type data in the fields provided and/or select structures by typing the number of the first occurrence you wish to work with.

PODATE : 971214

PONUMBER : 654321

LINEITEMS : Occurs 5 times

TOTALITEMS :

COMMENT :

Although you cannot make any changes to the document on this panel, you can move through the structures. See “Navigating within a Business Document” on page 13-13 for information on moving through the business document.

- When you are finished viewing the business document, press Exit (F3) to return to the Business Documents panel (DE01).

Recalling a Business Document from the Network

To recall a business document that was sent to the network, follow these steps:

- On the Business Documents panel (DE01), type **L** in the action column next to the business document you want to recall, and press Enter. The Network Acknowledgments: Recall panel (DE16) is displayed.
- Type **Y** if you want a recall acknowledgment, or **N** if you do not.

DE01		Business Documents		1 to 6 of 6	
A	Document Key	Trading Partner	Data Format ID	Status	Date
1	P0123456				
-	P0123456				
-	P01234568906				
-	P0654321				
-	13579				
-	654321				

DE16 Network Acknowledgments: Recall
 Do you want a recall acknowledgment? (Y/N) Y

DataInterchange attempts to recall the business document, then redisplay the Business Documents panel (DE01), with the business document's status updated to **RECALL REQUESTED**.

Receiving a Business Document

To receive a business document, follow these steps:

1. On the Business Documents panel (DE01), type **R** in the action column next to any business document, and press Enter. The Receive business document panel (DE12) is displayed.

DE01		Business Documents		1 to 6 of 6	
A	Document Key	Trading Partner	Data Format ID	Status	Date
r	P0123456				
-	P0123456				
-	P01234568906				
-	P0654321				
-	13579				
-	654321				

DE12 Receive Business Document
 From trading partner . . . _____
 Requestor ID _____
 Specific name _____

2. Complete the fields as follows:

- a. In the **From trading partner** field, type the name of the trading partner who sent the file. Leave this field blank to receive documents from all trading partners.
- b. In the **Requestor ID** field, type your requestor ID.
- c. In the **Specific name** field, type a name for a specific type of data. This field is optional. If your network does not support receiving documents by specific name, this field is not displayed.
- d. Press Enter.

All business documents meeting the selection criteria are received, placed in the Transaction Store, translated, and made available to you in the list of business documents. If any received document creates a functional acknowledgment, the functional acknowledgment is also placed in the Transaction Store.

Translating a Business Document to EDI Format

To translate a business document EDI standard format and put it in the Transaction Store, on the Business Documents panel (DE01), type **T** in the action column next to the business document you want to translate, and press Enter.

DataInterchange translates the business document, places it in the Transaction Store, then updates the status of the business document on the Business Document panel (DE01).

Working with Messages

A message is a short, free-form communication with a trading partner. A message can be up to 50 lines long and can contain any characters you can type on a keyboard.

When working with messages, you can do the following:

- Add messages
- Send messages to trading partners
- Copy messages
- Delete messages
- Print messages
- Update messages
- View messages
- Recall messages from the network
- Receive messages from trading partners
- Update the status of messages

Messages Panel (DM01)

When you work with messages, you select actions from the Messages panel (DM01). To access this panel, follow these steps:

1. From the Interactive Entry Facility Menu (MM01), select **Messages**. If there are no existing messages, the Select Message Action panel (DM15) is displayed.
2. To add a message, select option 1, then see step 2 under “Adding a Message” on page 13-28.
3. To receive a message, select option 2, then see step 2 under “Receiving a Message” on page 13-32.

DM15

Select Messages Action

No messages exist to which you have authority to access. Type the number of your selection and press Enter, or press F3 (Exit) to exit.

Choice====> 1. Add message
 2. Receive message

If there are existing messages, the Messages panel (DM01) is displayed.

Add	Copy	Delete	Print	Update	View	recall	Receive	Send	sStatus
DM01	Messages								1 to 1 of 1
A	Subject			Trading			Status		Date
_	Backorders			Partner			HELD		97/12/25
				PISCES					

The Messages panel (DM01) contains the following information:

Column	Description
Subject	The topic of the message.
Trading Partner	The nickname of the trading partner associated with this message.
Status	The status of the message. See "Message Status" for details.
Date	For an outbound message that has not been sent, the date of creation or the last update. For an outbound message that has been sent, the date the message was sent. For an inbound message, the date the message was received and translated.

- When you are finished working with messages, press Exit (F3) to return to the Interactive Entry Facility Menu (MM01).

Message Status

The following table describes the status values for messages. Some of these status values are similar to the transaction status codes described in "Transaction Status Codes" on page 10-28, but these values apply only to messages within IEF.

Table 13-4. Message Status

Status	Description
Held	The message is to be sent at a later time.
Send requested	DataInterchange successfully issued a send command to the network.
Sent to network	DataInterchange successfully sent a message to the network.
Recall requested	DataInterchange successfully issued a recall command.
Send request error	DataInterchange reported an error while attempting to issue a send command to the network.
Not sent-network error	The network encountered errors preventing the message from being sent.
Sent with errors	The network processed the send command with minor errors.
Accepted by network	DataInterchange received a network receipt acknowledgment.
Delivered by network	DataInterchange received a network delivery acknowledgment.
Purged by network	DataInterchange received a purge acknowledgment from the network.
Recalled	DataInterchange received a cancel acknowledgment from the network.
Received	The message was successfully received from the network.

Message Functions and Status

A message's status determines which functions are valid. The following table shows which functions are valid for each status value, where X indicates that the function is valid. For example, if a message's status is SEND REQUESTED, you can copy, delete, print, view, or recall the message, but you cannot update or send it.

Table 13-5. Message Functions and Status

Status	Delete	Update	View	Copy	Send	Recall	Print
Held	X	X	X	X	X		X
Send requested	X		X	X		X	X
Sent to network	X		X	X			X
Recall requested	X	X	X	X			X
Send request error	X	X	X	X	X		X
Not sent net error	X	X	X	X	X		X
Sent with errors	X		X	X		X	X
Accepted by network	X		X	X		X	X
Delivered by network	X		X	X			X
Purged by network	X	X	X	X	X		X
Recalled	X	X	X	X	X		X
Received	X		X				X

Adding a Message

To add a message, follow these steps:

1. On the Messages panel (DM01), type **A** in the action column next to any message, and press Enter. The Add Message panel (DM02) is displayed.

DM02

Add Message

1 to 13 of 50

Subject _____

Trading partner _____ +

Type your message on the following lines. Press Enter to end the message.

2. Complete the fields as follows:
 - a. In the **Subject** field, type the topic of this message.

- b. In the **Trading partner** field, type the nickname of the trading partner you are preparing this message for.
- c. On the message lines, type your message. If the message is longer than the number of lines you can see on the panel at one time, use F7 (Backward) and F8 (Forward) to scroll.
- d. Press Enter when you finish typing the message.

The Status Choices Menu (DM07) is displayed.

DM02	Add Message	1 to 13 of 50
Subject Christmas closings_____		
Trading partner PISCES_____		
This is the message f All stores will be cl _____ _____ _____ _____	<div style="text-align: center;">DM07 Status Choices</div> <p>Select the status you wish for this message.</p> <p>Choice ==> _ 1. Send now 2. Hold for send at a later time</p>	

3. To save and send the message now, select **Send now**, then go to step 2 under “Sending a Message.”
4. To save the message to send later, select **Hold for send at a later time**. DataInterchange saves the message, then redisplay the Messages panel (DM01).

Sending a Message

To send a message from the Messages panel (DM01), follow these steps:

1. Type **S** in the action column next to the message you want to send, and press Enter. The Network Acknowledgments Send panel (DM08) is displayed.

Only the acknowledgments that your network supports are shown on this panel. If your network does not support any acknowledgments, this panel is not displayed.

2. Type a slash (/) next to each type of network acknowledgment you want to receive, and press Enter.

DM01	Messages		1 to 1 of 1															
<table style="width: 100%; border-collapse: collapse;"> <tr> <th style="width: 10%;">A</th> <th style="width: 40%;">Subject</th> <th style="width: 20%;">Trading Partner</th> <th style="width: 20%;">Status</th> <th style="width: 10%;">Date</th> </tr> <tr> <td>s</td> <td>Christmas closing</td> <td>PISCES</td> <td>HELD</td> <td>97/12/03</td> </tr> <tr> <td>_</td> <td>Backorders</td> <td>PISCES</td> <td>SEND REQUESTED</td> <td>97/12/04</td> </tr> </table>	A	Subject	Trading Partner	Status	Date	s	Christmas closing	PISCES	HELD	97/12/03	_	Backorders	PISCES	SEND REQUESTED	97/12/04	<div style="text-align: center;">DM08 Network Acknowledgments: Send</div> <p>Place a '/' next to the acknowledgments you wish to receive and press Enter, or just press Enter if you wish to receive no acknowledgments.</p> <p>_ Accepted by network _ Delivery to trading partner _ Purged by network</p>		
A	Subject	Trading Partner	Status	Date														
s	Christmas closing	PISCES	HELD	97/12/03														
_	Backorders	PISCES	SEND REQUESTED	97/12/04														

The Network Access panel (DM11) is displayed.

DM01 Messages 1 to 1 of 1			
A	Subject	Trading Partner	Status
s	Christmas closing	PISCES	HELD
-	Backorders	PISCES	SEND REQUESTED

DM08 Network Acknowledgments: Send

DM11 Network Access

Requestor ID

Specific name

- Delivery to trading partner

- Purged by network

If your network does not support receiving documents by a specific name, the **Specific name** field is not displayed.

3. Complete the fields as follows:

- a. In the **Requestor ID** field, type your requestor ID.
- b. In the **Specific name** field, type the name for a specific type of data.
- c. Press Enter.

DataInterchange sends the message, then redisplay the Messages panel (DM01).

Copying or Updating a Message

Copy lets you create a new message by copying and changing an existing message.

To copy a message, on the Messages panel (DM01), type **C** in the action column next to the message you want to copy, and press Enter. The Copy Message panel (DM03) is displayed with the text from the message you copied. Except for the title, this panel is the same as the Add Message panel (DM02). The rest of the procedure is the same as in “Adding a Message” on page 13-28.

Update lets you change an existing message without creating a new message.

To update a message, on the Messages panel (DM01), type **U** in the action column next to the message you want to update, and press Enter. The Update Message panel (DM05) is displayed with the existing text. Except for the title, this panel is the same as the Add Message panel (DM02). The rest of the procedure is the same as in “Adding a Message” on page 13-28.

Deleting a Message

To delete a message, follow these steps:

1. On the Messages panel (DM01), type **D** in the action column next to the message you want to delete, and press Enter. The Delete Message panel (DM04) is displayed.
2. Press Enter to delete the message.

DM01	Messages	1 to 2 of 2
A Subject	Trading Partner	Status
d Backorders		Date
- Christmas closings		

DM04 Delete Message

Subject : Backorders
Trading partner . : PISCES

Press Enter to delete the message. Press the
Cancel key to cancel the request.

DataInterchange deletes the message, then redisplay the Messages panel (DM01).

Printing a Message

To print a message, on the Messages panel (DM01), type **P** in the action column next to the message you want to print, and press Enter.

DataInterchange sends the message data to the printer, print queue, or print file you specified at logon.

Viewing a Message

To view a message, follow these steps:

1. On the Messages panel (DM01), type **V** in the action column next to the message you want to view, and press Enter. The View Message panel (DM06) is displayed.

DM06	View Message	1 to 13 of 50
Subject	Christmas closings	
Trading partner . .	PISCES	
Status	HELD	
All stores will be closed Christmas weekend.		

2. Press F3 (Exit) when you are finished viewing the message.

Recalling a Message

To recall a message, follow these steps:

1. On the Messages panel (DM01), type **L** in the action column next to the message you want to recall, and press Enter. The Network Acknowledgments: Recall panel (DM12) is displayed.

2. Type **Y** if you want a recall acknowledgment, or **N** if you do not.

DM01		Messages		1 to 1 of 1	
A	Subject	Trading Partner	Status	Date	
1	Backorders	PISCES	SEND REQUESTED	97/12/04	
-	Christmas closing	PISCES	HELD	97/12/03	

DM12 Network Acknowledgments: Recall

Do you want a recall acknowledgment? (Y/N) Y

DataInterchange attempts to recall the message from the network, then redisplay the Messages panel (DM01).

Receiving a Message

To receive a message, follow these steps:

1. On the Messages panel (DM01), type **R** in the action column next to any message, and press Enter. The Receive Message panel (DM10) is displayed.

DM01		Messages		1 to 1 of 1	
A	Subject	Trading Partner	Status	Date	
r	Backorders	PISCES	SEND REQUESTED	97/12/04	
-	Christmas closing	PISCES	HELD	97/12/03	

DM10 Receive Message

From trading partner . . . _____ +

Requestor ID _____ +

2. Complete the fields as follows:
 - a. In the **From trading partner** field, type the name of the trading partner you want to receive messages from. If you do not specify a trading partner, you receive messages from all trading partners.
 - b. In the **Requestor ID** field, type your requestor ID.
 - c. Press Enter.

All messages meeting the selection criteria are received and added to the list on the Messages panel (DM01).

Updating the Status of Messages

To update the status of the messages, type **T** in the action column next to any message on the Messages panel (DM01), and press Enter.

DataInterchange retrieves status information from the network.

Handling File Requests

The file requests function is not available in DataInterchange for CICS.

IEF uses file requests to process and track the sending, receiving, and recalling of files. You can add, delete, or view file requests.

When working with file requests, you can:

- Add request to send a file
- Send a file
- Delete a file request
- View a file request
- Recall a file from the network
- Receive a file
- Update the status of file requests

Unlike business documents and messages, you do not enter files using IEF panels. DataInterchange only gives you a means of sending, recalling, and receiving files.

File Requests Panel (DF01)

When working with file requests, you select actions from the File Requests panel (DF01). To access this panel, follow these steps:

1. From the Interactive Entry Facility Menu (MM01), select **File requests**. If there are no existing file requests, the Select File Request Action panel (DF12) is displayed.

From this panel, you can add or receive a file request.

2. To add a file request, select option 1, then see step 2 under “Adding a Request to Send a File” on page 13-35.
3. To receive a file request, select option 2, then see step 2 under “Receiving a File” on page 13-38.

If there are existing file requests, the File Requests panel (DF01) is displayed.

```
DF12                      Select File Requests Action

No file requests exist. Choose one of the actions listed by typing the number
of your selection below and pressing Enter, or press the Exit key to exit.

Choice===>  _      1. Add file request
                  2. Receive file request
```

Add	Delete	View	recall	Receive	Send	sTatus
DF01				File Requests		1 to 1 of 1
A	File Name			Trading Partner	Status	Date
_	WORKERB.EDI.BACKORD			PISCES	HELD	97/12/27

The File Requests panel (DF01) contains the following information:

Column	Description
File name	The name of the file for which the request was created.
Trading partner	The name of the trading partner as specified in the trading partner profile.
Status	The status of this file request. See “File Request Status” for details.
Date	For an outbound file that has not been sent, the date the file request was created. For an outbound file that has been sent, the date the file was sent. For an inbound file, the date the file was received.

File Request Status

The following table describes the status values for file requests. Some of these status values are similar to the transaction status codes described in “Transaction Status Codes” on page 10-28, but these values apply only to file requests within IEF.

Table 13-6. File Request Status

Status	Description
Held	The file is to be sent at a later time.
Send requested	DataInterchange successfully issued a send command to the network.
Sent to network	DataInterchange successfully sent a message to the network.
Recall requested	DataInterchange successfully issued a recall command.
Send request error	DataInterchange reported an error while attempting to issue a send command to the network.
Not sent-network error	The network encountered errors that prevented processing of the send command.
Sent with errors	The network processed the send command with minor errors.
Accepted by network	DataInterchange received a network receipt acknowledgment.
Delivered by network	DataInterchange received a network delivery acknowledgment.
Purged by network	DataInterchange received a purge acknowledgment from the network.
Recalled	DataInterchange received a cancel acknowledgment from the network.
Received	The file was successfully received from the network.

File Request Functions and Status

The file request’s status determines which functions are valid. The following table shows which functions are valid for each status value, where X indicates that the function is valid. For example, if file request status is SEND REQUESTED, you can delete or print the file request, or recall the file, but you cannot send the file.

Table 13-7. File Request Functions and Status

Status	Delete	View	Send	Recall
Held	X	X	X	
Send requested	X	X		X
Sent to network	X	X		X
Recall requested	X	X		
Send request error	X	X	X	
Not sent net error	X	X	X	
Sent with errors	X	X		X
Accepted	X	X		X
Delivered	X	X		
Purged	X	X	X	
Recalled	X	X	X	
Received	X	X		

Adding a Request to Send a File

To add a file request, follow these steps:

1. On the File Requests panel (DF01), type **A** in the action column next to any item, and press Enter. The Add Request to Send a File panel (DF02) is displayed.

DF02
Add Request to Send a File

File name _____
Trading partner . . . _____

2. Complete the fields as follows:
 - a. In the **File name** field, type the name of the file to send.
 - b. In the **Trading partner** field, type the name of the trading partner you are sending the file to.
 - c. Press Enter.

The Status Choices Menu (DF05) is displayed.

DF02
Add Request to Send a File

File name WORKERB.EDI.BACKORD
Trading partner . . . PISCES

DF05
Status Choices

Select the status you wish for this request.

Choice ==> _ 1. Send the file now
2. Hold the request to send

3. To send the file now, select **Send the file now**, then go to step 2 under “Sending a File” on page 13-36.
4. To hold the file, select **Hold the request to send**. DataInterchange saves the file request, then redisplay the File Requests panel (DF01).

Sending a File

If you selected **Send the file now** from the Status Choices Menu (DF05), start at step 2.

To send a file from the File Requests panel (DF01), follow these steps:

1. Type **S** in the action column next to the file you want to send, and press Enter. The Network Acknowledgments panel (DF06) is displayed.

Only the acknowledgments that your network supports are shown on this panel. If your network does not support any acknowledgments, this panel is not displayed.

2. Type a slash (/) next to each type of network acknowledgment you want to receive, and press Enter.

DF01		File Requests		1 to 1 of 1	
A	File Name	Trading Partner	Status	Date	
s	WORKERB.EDI.BACKORD	PISCES	SEND REQUESTED	97/12/03	

DF06 Network Acknowledgments: Send

Place a '/' next to the acknowledgments you wish to receive and press Enter, or just press Enter if you wish to receive no acknowledgments.

- _ Accepted by network
- _ Delivery to trading partner
- _ Purged by network

The Network Access panel (DF08) is displayed.

DF01		File Requests		1 to 1 of 1	
A	File Name	Trading Partner	Status	Date	
s	WORKERB.EDI.BACKORD	PISCES	SEND REQUESTED	97/12/03	

DF06 Network Acknowledgments: Send

DF08 Network Access

Requestor ID _____

Specific name . . . _____

- _ Purged by network

If your network does not support receiving documents by a specific name, the **Specific name** field is not displayed.

3. Complete the fields as follows:

- In the **Requestor ID** field, type your requestor ID.
- In the **Specific name** field, type the name for a specific type of data.
- Press Enter.

DataInterchange sends the file, then redisplay the File Requests panel (DF01).

Deleting a File Request

To delete a file request, follow these steps:

- On the File Requests panel (DF01), type **D** in the action column next to the file request that you want to delete, and press Enter. The Delete File Request panel (DF03) is displayed.
- Press Enter to delete the file request.

DF01	File Requests	1 to 1 of 1
A	File Name	Trading Partner
d	WORKERB.EDI.B	Status
		Date

DF03 Delete File Request

Trading partner . : PISCES

File name : WORKERB.EDI.BACKORD

Press Enter to delete the request or press the Cancel key to keep the request.

DataInterchange deletes the file request, then redisplay the File Requests panel (DF01).

Viewing a File Request

To view a file request, follow these steps:

- On the File Requests panel (DF01), type **V** in the action column next to the file request you want to view, and press Enter. The View File Request panel (DF04) is displayed.

DF04	View File Request
File name :	WORKERB.EDI.BACKORD
Trading partner . :	PISCES
Status :	HELD
Specific name . . :	

You can look at your file request, but you cannot make any changes. An advantage of this panel is that you can see the complete name of the trading partner. On the File Requests panel (DF01), the trading partner name is truncated to 8 characters.

- When you are finished viewing the file request, press F3 (Exit) to return to the File Requests panel (DF01).

Recalling a File

To recall a file request, follow these steps:

1. On the File Requests panel (DF01), type **L** in the action column next to the file request you want to recall, and press Enter. The Network Acknowledgments Recall panel (DF09) is displayed.
2. Type **Y** if you want a recall acknowledgment, or **N** if you do not.

DF01	File Requests			1 to 1 of 1
A	File Name	Trading Partner	Status	Date
1	WORKERB.EDI.BACKORD	PISCES	SEND REQUESTED	97/12/03

DF09 Network Acknowledgments: Recall

Do you want a recall acknowledgment? (Y/N) Y

DataInterchange attempts to recall the file, then redisplay the File Requests panel (DF01).

Receiving a File

To receive a file, follow these steps:

1. On the File Requests panel (DF01), type **R** in the action column next to any file request, and press Enter. The Receive File panel (DF07) is displayed.

DF01	File Requests			1 to 1 of 1
A	File Name	Trading Partner	Status	Date
r	WORKERB.EDI.BACKORD	PISCES	SEND REQUESTED	97/12/03

DF07 Receive File

From trading partner . _____

Requestor ID _____

Specific name _____

Receive to file name . _____

2. Complete the fields as follows:

- a. In the **From trading partner** field, type the name of the trading partner who sent the file. Leave this field blank to receive files from all trading partners.
- b. In the **Requestor ID** field, type your requestor ID.
- c. In the **Specific name** field, type a name for a specific type of data. This field is optional. If your network does not support receiving files by specific name, this field is not displayed.
- d. In the **Receive to file name** field, type the name of the file you want to receive files to.
- e. Press Enter.

If a file meets the selection criteria, DataInterchange writes it to the output file name, then redisplay the File Requests panel (DF01).

Updating the Status of File Requests

To update the status of the file requests, type **T** in the action column next to any file request on the File Requests panel (DF01), and press Enter.

DataInterchange retrieves status information from the network.

Chapter 14. Customizing Business Documents

Starting the Document Layout Utility	14-1
Input to the Document Layout Utility	14-2
The Application Data Format	14-2
The Layout Statements	14-2
Syntax Rules	14-2
LAYOUT	14-3
GROUP	14-3
ENDGROUP	14-4
STRUCT	14-5
LITERAL	14-5
FIELD	14-5
Output from the Document Layout Utility	14-7
Printing Customized Documents in Batch	14-7
Document Layout Example	14-7
The Sample Application Data Format	14-7
The Sample Purchase Order	14-8
Including IEF Control Fields in the Layout	14-9
Suppressing Individual Fields	14-9
Defining Fields that Require More than One Line	14-10
Suppressing Groups of Fields and Literals	14-11
Defining Repeating Structures	14-12
Printing Page Headings	14-13
The Complete Layout Statement File	14-14
Predefined Layouts	14-16
The Purchase Order Layout	14-16
The Invoice Layout	14-18
Document Layout Messages	14-20

Chapter 14. Customizing Business Documents

A *customized document* is a transaction, such as a purchase order, that you can view or print in a familiar format. You can use the *View* and *Print* actions of the Interactive Entry Facility (IEF), or use the PERFORM PRINT CUSTOM LAYOUT command to print the document. Before you can do this, you must create a *document layout* for each type of transaction you want to view or print. A document layout describes, by row and column, how you want your data to display on a view panel, or print on a printout.

To customize a document, you must do the following:

1. Create an application data format (see Chapter 7, “Defining and Working With Your Application Data”).
2. Define the transaction as a document in IEF (see Chapter 13, “Using the Interactive Entry Facility (IEF)”).
3. Use the Document Layout Utility to create a document layout (explained in this chapter).

Your input to the Document Layout Utility is a series of *layout statements* in a text file. The Document Layout Utility uses the statements to build a document layout record. If a record for the layout already exists in the database, the utility replaces it with the new record. If a record does not exist, the Document Layout Utility adds it to the database.

Starting the Document Layout Utility

Note: The Document Layout Utility is not available in CICS.

To start the Document Layout Utility for DB2, type **EDILTDB2** at the TSO ready prompt. For VSAM, type **EDILTVSM**. Verify with the person who installed DataInterchange for you that these commands for starting the utility were not changed during installation.

These commands execute a CLIST that prompts you for the name of the layout statement file, and creates a print file named *userid.EDI.SIDPRT* if one does not exist (where *userid* is your TSO user ID). Let the CLIST create the output data set to ensure that the proper file attributes are used.

You can include the following options with the commands:

- Language profile (LANGPROF)
- Language ID (LANGID)
- System ID (SYSID)
- Print file (SIDPRT)

For descriptions of LANGPROF and LANGID, see “Language Profile (LANGPROF)” on page 4-3. For a description of SYSID, see “Logon Options for DataInterchange for MVS” on page 2-1. The SIDPRT option allows you to override the default name of *userid.EDI.SIDPRT* for the print file. For example:

```
EDILTDB2 LANGPROF(ENU) LANGID(ENU) SYSID(TEST)
```

You may want to edit the CLIST to eliminate entering options each time you start the Document Layout Utility.

Input to the Document Layout Utility

To create a document layout, the utility uses an existing application data format and an existing text file of layout statements. The records in this file have a fixed length of 80 characters. These statements inform the Document Layout Utility which fields in the application data format are used by the document layout. They also contain the literals that you want to appear in the document. Literals can provide titles, column headings, prompts, and even values for fields in the data format. One field or literal cannot overlap another in the document layout.

The Application Data Format

An application data format describes the application data for one type of transaction, such as a purchase order. The application data format must be available to the utility, and the person writing the layout statements must be familiar with the application data format. The utility uses the application data format to verify the layout statements; it does not modify the data format definition.

You can use the document layout to initialize fields defined in the application data format. The initialized fields do not have to be visible on the document layout, and not every field in an application data format has to be mapped to the document layout.

The Layout Statements

The six layout statements are:

- LAYOUT
- GROUP
- ENDGROUP
- STRUCT
- LITERAL
- FIELD

Each layout statement is described following “Syntax Rules.” For examples of using the layout statements, see “Document Layout Example” on page 14-7.

Syntax Rules

The following syntax rules apply when you write layout statements:

- Each input file can contain layout statements for one application data format only.
A partitioned data set with a member for each document layout is a convenient way to manage the files.
- Each statement must be on a line by itself.
You can use blank lines to improve readability.
- Each line must begin with a statement name (for example, LAYOUT).
- Parameters are enclosed in parentheses.
- Blanks can appear anywhere in a line except within a statement name, structure name, or field name.
You can use indenting and spacing to improve readability.
- If necessary, use the space after the closing parenthesis for comments, eye-catchers, or unique character strings that are easy to locate.

- The utility is not case sensitive; use mixed case if desired.
- To omit an optional parameter, use a comma (,) in its position.
- To use the default value for a parameter, use a 0 in its position.
- Row values can be specified as absolute values or as offsets (increments or decrements) from the current output line number. Row offsets are specified as a plus (+) or minus (-) preceding a value (for example, +1). A 0 means the current line number. Each row specified resets the current row number.
- Use offsets rather than absolute row and depth values whenever possible. Offsets make the layout easier to create and maintain.

LAYOUT

The LAYOUT statement identifies the application data format and indicates whether the layout is for printing, viewing, or both. This is the first statement in each file, and it appears only once in a file.

The LAYOUT syntax is:

```
LAYOUT ( data format, layout type, page depth, layout width )
```

where:

data format

specifies the application data format for the layout that is being constructed.

layout type

specifies the layout of the document, either PRINT, VIEW, or BOTH, depending upon the intended use of the layout. When you PRINT or VIEW a document in IEF, the program first searches for a layout with a type that matches the request. If there is no match, it looks for type BOTH.

page depth

specifies the number of rows in a page for a print layout and is ignored for a VIEW layout. Set this parameter to 0 to use the default depth of 60.

layout width

specifies the width of a page for a PRINT layout and is ignored for a VIEW layout. This parameter must be set to 79.

GROUP

The GROUP statement associates any number of fields and literals. Fields and literals are grouped to:

- Provide heading information for printed layouts
- Improve clarity and readability
- Suppress the printing or displaying of fields that contain no data
- Print or display multiple occurrences of a structure while giving layout information only for the first occurrence

Groups can be nested within other groups.

Customizing Business Documents

The GROUP syntax is:

```
GROUP ( group name, group type, row, column, depth, width, repeat )
```

where:

group name

supplies a name or label for a group. It provides clarity and is optional.

group type

is one of the following:

Type	Description
HEADING	Specifies a group that prints at the top of each page when printing a document.
SUPPRESS	Specifies a group that is omitted from the display or page if there is no data for the fields in that group. A field is considered to have no data if its value is blanks or zeros. If all fields on a line of the layout are suppressed, that line is omitted from the output, and lines below it are moved up. When suppression occurs, fields are not shifted horizontally, and lines are not merged. When a group is nested within another group, the outer group is not considered to have data if the inner group contains data.
DEFINE	Specifies a group that provides a description of fields and literals.

row, *column*, *depth*, *width*

provides values for fine-tuning the appearance of a document. One example is to exclude one or more blank lines that follow a suppressed group. Use zeros to instruct the utility to calculate the group's boundary. You can use a positive offset value for the depth parameter to insert blank lines at the end of a group. For example, using +3 causes the utility to add 3 to the calculated group depth. Using an offset value makes it unnecessary for you to calculate the group depth and produces a layout that is easier to maintain.

repeat

produces successive occurrences of fields and literals in a group that is mapped to a repeating structure. The repetitions begin with the occurrence given in a following STRUCT statement. Each occurrence occupies the least number of lines needed to contain the data, as calculated from the literals and fields in the group or as set by the depth parameter.

ENDGROUP

The ENDMETHOD statement closes the current group. When a group is closed, the current line number is set to the last row of the last occurrence in that group.

The ENDMETHOD syntax is:

```
ENDGROUP ( group name )
```

where:

group name

is the name of the group. This optional parameter provides clarity.

STRUCT

The STRUCT statement provides the path to the correct fields in an application data format. For example, a STRUCT statement precedes the quantity, price, and other fields in a line item structure. You must include as many STRUCT statements as needed to lead from the base structure to the desired inner structure. Once the target structure is identified, it remains the target until you establish another structure as the one with which you are working. When a STRUCT statement is not immediately preceded by another STRUCT statement, the structure pointer is reset to the base structure and the new path must be defined from that point.

If the structure referred to has multiple occurrences (maximum use is greater than 1), the statement must include an occurrence number.

The STRUCT syntax is:

```
STRUCT ( structure name, occurrence number )
```

where:

structure name

specifies the name of a structure, other than the base structure, defined by the application data format. The search for any field always begins at the base structure. You do not have to include it in the search path.

occurrence number

indicates the occurrence of the structure you are working with.

LITERAL

The LITERAL statement supplies a literal value that appears in the layout.

The LITERAL syntax is:

```
LITERAL ( row, column, text )
```

where:

row, *column*

specifies the position of the literal text on the layout. The row can be specified as an absolute number or as an offset from the current row number.

text

contains the literal text exactly as it appears on the layout. The utility removes leading and trailing blanks. The literal text can include any number of left parentheses, but only one right parenthesis.

FIELD

The FIELD statement names a field in the data format that is either to appear on the document or to be given an initial value, or both.

The FIELD syntax is:

```
FIELD ( field name, row, column, depth, width, suppress, initial )
```

Customizing Business Documents

where:

field name

specifies the name of a field in the current structure of the application data format, or the name of a control field used by IEF. The following list describes the control fields that format data in the document layout. To obtain the value of a control field, use its name as the *field name* parameter. All control field names begin with an underscore.

Control field	Description
_PAGENUM	Page number (printing only)
_TOTPAGES	Total pages (printing only)
_PRTDATE	Print date (printing only)
_PRTTIME	Print time (printing only)
_DATAFORMAT	Data format ID
_TRADINGPARTNER	Trading partner nickname
_INTTRADEPART	Internal trading partner ID
_REQUESTOR	Requestor ID
_DOCDATE	Document date
_STATUS	Document status (INCOMPLETE, READY, HELD, and so forth)
_SPECIFICNAME	Specific name (a value used for receiving selected documents)
_RECEIPT	Network receipt indicator
_DELIVERY	Network delivery indicator
_PURGE	Network purge indicator

row, column

modifies the position of the field if it is to appear on the document. You can specify the row as an absolute value or an offset. If a layout position is not given, the *initial* parameter must be present.

depth

does nothing at this time.

width

specifies the maximum length of a line. This value is used for splitting the data for the field over two or more lines. It applies only to fields with a data type of AN or CH. The utility splits the data into equal parts by dividing the field length in the application data format by the width. The last part is shorter than the others if the field length is not evenly divisible by the width. Each field part appears on successive lines in the position specified by the column parameter.

suppress

specifies whether the field is removed from the page or display if the field has no data. Valid values are:

Value	Description
0	Do not remove it
1	Remove it

You can suppress fields in a group.

initial

supplies an initial value. This parameter can be used only in fields with a data type of ID, CH, or AN. If the data type is AN, the value must contain alphanumeric characters only. For the other types, the utility accepts any entry.

If you supply an initial value but no row and column position, the field is set to that value but does not appear on the document layout. You can use this combination, for example, to initialize fields that are required by the standard or the receiver but not entered on a data entry panel.

If you provide an initial value and a column and row, the field appears on the document layout at the given position with the initial value. This initial value also appears on the IEF Add, Copy, and Update document entry panels as the contents of the field. Note that the initial value can be a code that is used to look up the actual data in a table associated with the field. The layout example shown later uses this technique to initialize headings for name and address fields. The utility does not check the table for a value that matches the initial value. That check occurs during translation.

Output from the Document Layout Utility

The Document Layout Utility produces a document layout (DOCMAP) record and a print file.

IEF uses the DOCMAP record to set initial values when a document is entered or copied, and to format a document for viewing or printing.

The print file, named *userid.EDI.SIDPRT*, contains *pictures* of the layout, enabling you to see the results without running IEF. One picture suppresses all fields that you marked for suppression. (If you requested no suppression, this picture is omitted.) A second picture shows the entire layout. It is identical to the view you would get through IEF except that underscores appear in place of data. The utility provides display literals with underscores so you can see how long the fields are.

If the utility finds errors in your layout statements, the print file contains error messages rather than a layout picture. These messages are also displayed during processing. See “Document Layout Messages” on page 14-20 for details.

Each time you run the utility, it overlays the contents of the print file.

Printing Customized Documents in Batch

When running the Interactive Entry Facility, you can use the *View* and *Print* actions to see the results of your customized layouts. You can also use a batch process to print customized documents. For more information, see *DataInterchange Programmer's Reference*.

Document Layout Example

This section uses a sample purchase order to illustrate uses of the Document Layout Utility.

The Sample Application Data Format

A document layout must refer to an existing application data format. The following is the application data format for the example:

Customizing Business Documents

Field/Structure name	Type	Length	Max Use
BASESTRUCTURE	ST		1
HEADER	ST		1
PONUMBER	AC	12	
PODATE	DT	6	
NOTE	ST		2
NOTETEXT	CH	80	
NAMEANDADDRESS	ST		4
ORGANIZATIONID	ID	2	
NAME	ST		1
ORGNAME1	CH	18	
ORGNAME2	CH	18	
ORGIDCODE	CH	12	
ADDRESS	ST		1
ORGADDR1	CH	18	
ORGADDR2	CH	18	
ORGCITYSTATEZIP	CH	18	
LINEITEMS	ST		25
ITEMNUMBER	AN	5	
PRODUCTID	AN	13	
DESCRIPTION	CH	28	
QUANTITY	R	4	
UNITOFMEASURE	ID	2	
UNITPRICE	R2	10	
TRAILER	ST		1
POTOTAL	R2	13	

The Sample Purchase Order

The following is the sample purchase order. Underscores appear where blank fields are not suppressed.

PO Number: 847620	Purchase Order	Page 1 of 1			
PO Status: RECEIVED		Printed 03/30/98			
Bill to	Ship to				
The A Company	Company B				
<u>1047 Main Street</u>	<u>1257 Broadstreet</u>				
<u>Anytown ST 12345</u>	<u>Suite 107</u>				
	Hometown ST 54321				
Note: ***** PLEASE EXPEDITE PROCESSING OF THIS PURCHASE ORDER !!! *****					
Note: _____					
Item	Product ID	Description	Qty	UM	Unit Price
00001	2C2-4224		4	LB	15.99
00002	198-5664	A few words about this thing	1	LB	4.99
_____	_____	_____	_____	_____	_____
_____	_____	_____	_____	_____	_____
_____	_____	_____	_____	_____	_____
_____	_____	_____	_____	_____	_____
PO Total:					68.95

The following sections use this purchase order to demonstrate the following features you can include in your document layouts.

- Including IEF control fields in the layout
- Suppressing individual fields
- Defining fields that require more than one line
- Suppressing groups of fields and literals
- Defining repeating structures
- Defining page headings

Including IEF Control Fields in the Layout

The following sample purchase order uses IEF control fields in the layout:

P0 Number: 847620	Purchase Order	Page 1 of 1
P0 Status: RECEIVED		Printed 03/30/98
Bill to	Ship to	

The utility gets the data associated with the literal **P0 Status:** from the same source in IEF that is used to display status information for documents listed on the Business Documents panel. The page numbers and print date are also control fields in IEF.

The layout statements for these control fields are:

```
Field ( _PAGENUM, 1, 66, 0, 0, 0, )
Field ( _TOTPAGES, 0, 73, 0, 0, 0, )
Field ( _STATUS, +1, 13, 0, 0, 0, )
Field ( _PRTDATE, 0, 69, 0, 0, 0, )
```

Suppressing Individual Fields

The application data format provides two lines for the company name and two lines for the street address. The **Bill to** information requires only one line for each. The **Ship to** information uses only one line for the company name, but it uses two lines for the address.

Setting the *suppress* parameter in a FIELD statement to 1 removes that field when it contains no data. Suppression occurs in two steps. First, the underscores that mark an empty field are blanked out. Second, if no other data occurs on the line, the entire line is removed, and all lines below it are moved up.

P0 Number: 847620	Purchase Order	Page 1 of 1
P0 Status: RECEIVED		Printed 03/30/98
Bill to	Ship to	
The A Company	Company B	
1047 Main Street	1257 Broadstreet	
	Suite 107	
Anytown ST 12345	Hometown ST 54321	
Note: ***** PLEASE EXPEDITE PROCESSING OF THIS PURCHASE ORDER !!! *****		
<hr/>		

Customizing Business Documents

All of the name and address fields are marked for suppression. Because neither company uses the second name field, the underscores for those fields are removed. Furthermore, because that line contains suppressed output and no other data occurs on the line, the entire line is removed, and the lines below it are shifted up. The underscores for the second address field for the **Bill to** company are also suppressed. But because that line contains data (**Suite 107**), it is not removed.

The following are the layout statements for the **Bill to** name and address. The initial value BT is a code for retrieving the heading **Bill to** from a table.

```
Struct ( HEADER )
Struct ( NAMEANDADDRESS, 1 )
  Field ( OrganizationID, 0, 2, 0, 0, 0, BT )
```

```
Struct ( HEADER )
Struct ( NAMEANDADDRESS, 1 )
Struct ( NAME )
  Field ( OrgName1, +1, 4, 0, 0, 0, )
  Field ( OrgName2, +1, 4, 0, 0, 1, )
```

```
Struct ( HEADER )
Struct ( NAMEANDADDRESS, 1 )
Struct ( ADDRESS )
  Field ( OrgAddr1, +1, 4, 0, 0, 0, )
  Field ( OrgAddr2, +1, 4, 0, 0, 1, )
  Field ( OrgCityStateZip, +1, 4, 0, 0, 0, )
```

Notice the following in the layout statement:

- STRUCT statements are used to establish a path to the correct fields
- FIELD statements are offset to show nesting
- Blank lines are used between structures

Defining Fields that Require More than One Line

You can split an alphanumeric or character field across several lines. A split field continues at the same column position for as many lines as necessary to accommodate the length of the field. You accomplish this by setting the *width* parameter to the number of characters you want on a line. In this example, the width of the **Note**: which has a length of 80 in the application data format, is set to 72. As a result, the last eight characters of the note appear on a second line (shown as underscores, because there is no data in these positions).

PO Number: 847620	Purchase Order	Page 1 of 1
PO Status: RECEIVED		Printed 03/30/98
Bill to	Ship to	
The A Company	Company B	
1047 Main Street	1257 Broadstreet	
	Suite 107	
Anytown ST 12345	Hometown ST 54321	
Note: ***** PLEASE EXPEDITE PROCESSING OF THIS PURCHASE ORDER !!! *****		

Adding field suppression to multiline fields produces only the lines needed to accommodate the nonblank portion of the field. In this example, suppressing the *NoteText* field in the first note removes the second

line of the note, because it is blank. The second note illustrates that no formatting occurs when lines are split.

PO Number: 847620 PO Status: RECEIVED	Purchase Order	Page 1 of 1 Printed 03/30/98
Bill to The A Company 1047 Main Street <hr/> Anytown ST 12345	Ship to Company B 1257 Broadstreet Suite 107 Hometown ST 54321	
Note: ***** PLEASE EXPEDITE PROCESSING OF THIS PURCHASE ORDER !!! *****		
Note: See Mrs. Jones for detail - Fast Forward Division, Dept. A1B, office 2 3E x6278		

The following are the layout statements for the previous example:

```
Struct ( HEADER )
Struct ( NOTE, 1 )
  Literal( 0, 2, Note: )
  Field ( NoteText, 0, 8, 0, 72, 1, )
```

```
Struct ( HEADER )
Struct ( NOTE, 2 )
  Literal( +2, 2, Note: )
  Field ( NoteText, 0, 8, 0, 72, 1, )
```

Suppressing Groups of Fields and Literals

A literal often precedes a field to explain the contents of the field (for example, **PO Number:**). If a field or several fields are to be suppressed, the associated literal text should also be removed. You do this with a **GROUP** statement, which puts the fields and literals into a logical unit.

Group suppression is not automatic. You request it by setting the *group type* parameter to **SUPPRESS**. If none of the fields within the group contains data, the group is blanked out, and any lines covered by the group that do not contain data are removed from the display or page. If any field in the group contains data, fields without data are shown as underscores. You can remove the underscores with field suppression.

PO Number: 847620 PO Status: RECEIVED	Purchase Order	Page 1 of 1 Printed 03/30/98
Bill to The A Company 1047 Main Street <hr/> Anytown ST 12345	Ship to Company B 1257 Broadstreet Suite 107 Hometown ST 54321	
Note: ***** PLEASE EXPEDITE PROCESSING OF THIS PURCHASE ORDER !!! *****		
Item	Product ID	Description Qty UM Unit Price

Each note is defined as a group whose group type is **SUPPRESS**. (The next section shows how these can be combined into one group with the *repeat* parameter.) Because the second note has no data, both

Customizing Business Documents

the *NoteText* field and the associated literal are suppressed. In the first note, field suppression removes the underscores from the second line.

The *depth* parameter for the note group contains an offset of +1. This results in a blank line following each unsuppressed note. The blank line makes the document more readable. The following is the layout statements for the previous example:

```
Group ( Notes, Suppress, +1, 0, +1, 0, 0 )
  Struct ( HEADER )
  Struct ( NOTE, 1 )
    Literal( 0, 2, Note: )
    Field ( NoteText, 0, 8, 0, 72, 1, )
  EndGroup ( Notes )
```

```
Group ( Notes, Suppress, +1, 0, +1, 0, 0 )
  Struct ( HEADER )
  Struct ( NOTE, 2 )
    Literal( 0, 2, Note: )
    Field ( NoteText, 0, 8, 0, 72, 1, )
  EndGroup ( Notes )
```

Defining Repeating Structures

Application data formats often contain multiple occurrences of a structure (maximum use greater than one). Structures for addresses and line items are examples. When defining a document layout, you can map the first occurrence and provide a repeat count to indicate how many times the fields and literals occur.

PO Number: 847620	Purchase Order	Page 1 of 1			
PO Status: RECEIVED		Printed 03/30/98			
Bill to	Ship to				
The A Company	Company B				
1047 Main Street	1257 Broadstreet				
	Suite 107				
Anytown ST 12345	Hometown ST 54321				
Note: ***** PLEASE EXPEDITE PROCESSING OF THIS PURCHASE ORDER !!! *****					
Note: See Mrs. Jones for detail - Fast Forward Division, Dept. A1B, office 2 3E x6278					
Item	Product ID	Description	Qty	UM	Unit Price
00001	2C2-4224		4	LB	15.99
00002	198-5664	A few words about this thing	1	LB	4.99
PO Total:					68.95

In this example, the second occurrences of the **Note:** and line item are produced from a definition of one occurrence. The **Bill to** and **Ship to** name and address sections can also be two occurrences of one structure.

The following are the layout statements for the previous example:

```
Group ( Notes, Suppress, +1, 0, +1, 0, 2 )
  Struct ( HEADER )
  Struct ( NOTE, 1 )
    Literal( 0, 2, Note: )
    Field ( NoteText, 0, 8, 0, 72, 1, )
  EndGroup ( Notes )

Group ( LineItems, Suppress, +1, 0, 0, 0, 6 )
  Struct ( LINEITEMS, 1 )
    Field ( ItemNumber, 0, 2, 0, 0, 0, )
    Field ( ProductID, 0, 8, 0, 0, 0, )
    Field ( Description, 0, 22, 0, 0, 0, )
    Field ( Quantity, 0, 50, 0, 0, 0, )
    Field ( UnitOfMeasure, 0, 55, 0, 0, 0, )
    Field ( UnitPrice, 0, 58, 0, 0, 0, )
  EndGroup ( LineItems )
```

You do not have to map all occurrences of the data format structure to the output. This example maps all occurrences (2) of the note structure to the output, but not all occurrences (25) of the line item structure. In addition, you are not required to begin mapping with the first occurrence of a structure.

Printing Page Headings

You can use another type of group to define a report heading to appear at the top of every printed page. Set the *group type* parameter to **HEADING** for this use. This group can contain any number of fields and literals. It can contain data from fields defined by the data format, as well as IEF control fields. If the layout type is **BOTH** (meaning that the layout can be used to either print or view the document), the heading does not appear when you view the document.

The heading does not have to include everything on the line. If the heading group defined only the page and print date, it would resemble the following when printed:

P0 Number: 847620	Purchase Order	Page 1 of 1
P0 Status: RECEIVED		Printed 03/30/98
Bill to	Ship to	

The same heading group would resemble the following when displayed on a terminal:

P0 Number: 847620	Purchase Order
P0 Status: RECEIVED	
Bill to	Ship to

The *page depth* parameter in the **LAYOUT** statement controls page breaks.

Customizing Business Documents

The following are the layout statements for the previous example:

```
Layout ( LayoutToolExample, Both, 55, 0 )

Literal ( 1,31, Purchase Order )
Literal ( 0, 2, PO Number: )
Struct ( HEADER )
    Field ( PONumber, 0, 13, 0, 0, 0, )
    Literal( +1, 2, PO Status )
    Field ( _STATUS, 0, 13, 0, 0, 0, )

Group ( RptHeading, Heading, 1, 0, 3, 0, 0 )
    Literal( 0, 61, Page )
    Field ( _PAGENUM, 0, 66, 0, 0, 0, )
    Literal( 0, 70, of )
    Field ( _TOTPAGES, 0, 73, 0, 0, 0, )
    Literal(+1, 61, Printed )
    Field ( _PRDATE, 0, 69, 0, 0, 0, )
EndGroup ( RptHeading )
```

The Complete Layout Statement File

The following is the complete list of layout statements for the sample purchase order:

```
Layout ( LayoutToolExample, Both, 0, 0 )

Literal ( 1,31, Purchase Order )
Literal ( 0, 2, PO Number: )
Struct ( HEADER )
    Field ( PONumber, 0, 13, 0, 0, 0, )
    Literal( +1, 2, PO Status )
    Field ( _STATUS, 0, 13, 0, 0, 0, )

Group ( RptHeading, Heading, 1, 0, 3, 0, 0 )
    Literal( 0, 61, Page )
    Field ( _PAGENUM, 0, 66, 0, 0, 0, )
    Literal( 0, 70, of )
    Field ( _TOTPAGES, 0, 73, 0, 0, 0, )
    Literal(+1, 61, Printed )
    Field ( _PRDATE, 0, 69, 0, 0, 0, )
EndGroup ( RptHeading )
```

```

Group ( Name1, Define, 4, 0, +1, 0, 0 )
  Struct ( HEADER )
  Struct ( NAMEANDADDRESS, 1 )
    Field ( OrganizationID, 0, 2, 0, 0, 0, BT )

  Struct ( HEADER )
  Struct ( NAMEANDADDRESS, 1 )
  Struct ( NAME )
    Field ( OrgName1, +1, 4, 0, 0, 0, )
    Field ( OrgName2, +1, 4, 0, 0, 1, )

  Struct ( HEADER )
  Struct ( NAMEANDADDRESS, 1 )
  Struct ( ADDRESS )
    Field ( OrgAddr1, +1, 4, 0, 0, 0, )
    Field ( OrgAddr2, +1, 4, 0, 0, 1, )
    Field ( OrgCityStateZip, +1, 4, 0, 0, 0, )
EndGroup ( Name1 )

Group ( Name2, Define, 4, 0, +1, 0, 0 )
  Struct ( HEADER )
  Struct ( NAMEANDADDRESS, 2 )
    Field ( OrganizationID, 0, 31, 0, 0, 0, ST )

  Struct ( HEADER )
  Struct ( NAMEANDADDRESS, 2 )
  Struct ( NAME )
    Field ( OrgName1, +1, 33, 0, 0, 0, )
    Field ( OrgName2, +1, 33, 0, 0, 1, )

  Struct ( HEADER )
  Struct ( NAMEANDADDRESS, 2 )
  Struct ( ADDRESS )
    Field ( OrgAddr1, +1, 33, 0, 0, 0, )
    Field ( OrgAddr2, +1, 33, 0, 0, 1, )
    Field ( OrgCityStateZip, +1, 33, 0, 0, 0, )
EndGroup ( Name2 )

Group ( Notes, Suppress, +1, 0, 3, 0, 2 )
  Struct ( HEADER )
  Struct ( NOTE, 1 )
    Literal( 0, 2, Note: )
    Field ( NoteText, 0, 8, 0, 72, 1, )
EndGroup ( Notes )

Literal ( +1, 2, Item Product ID      Description )
Literal ( 0, 50, Qty  UM Unit Price )

```

Customizing Business Documents

```
Group ( LineItems, Suppress, +1, 0, 0, 0, 6 )
  Struct ( LINEITEMS, 1 )
    Field ( ItemNumber, 0, 2, 0, 0, 0, )
    Field ( ProductID, 0, 8, 0, 0, 0, )
    Field ( Description, 0, 22, 0, 0, 0, )
    Field ( Quantity, 0, 50, 0, 0, 0, )
    Field ( UnitOfMeasure, 0, 55, 0, 0, 0, )
    Field ( UnitPrice, 0, 58, 0, 0, 0, )
  EndGroup ( LineItems )

Literal ( +2, 45, PO Total: )
Struct ( TRAILER )
  Field ( POTotal, 0, 55, 0, 0, 0, )
```

Predefined Layouts

DataInterchange includes predefined layouts for send and receive versions of a purchase order and an invoice. You can use the predefined layouts:

- As a model for developing customized layouts for documents defined by the EDI standards.
- Just as they are. Because the application data formats and trading partner transactions are already defined, your setup work is minor.

The Purchase Order Layout

Figure 14-1 on page 14-17 is an example of a purchase order as it might be displayed using the predefined document layout. To view or print a document whose application data format has been mapped to a layout, select *Business documents* from the Interactive Entry Facility Menu (MM01). Then select *View* or *Print* for any document that uses that application data format. To create a formatted document in a file, select *File*, then *File formatted*.

The following are associated with the predefined purchase order:

Item	Description
Data format IDs	GENPO (for sending) and GENPORCV (for receiving). You can view or print these application data formats after selecting <i>Application Data Formats</i> from the Administrator's Menu (MP01).
Transaction IDs	GENPOMAP (for sending) and GENPORCVMAP (for receiving). You can print the definitions of these transactions after selecting <i>Trading Partner Transactions</i> from the Administrator's Menu (MP01).
Tables	The names of the tables used in mapping the purchase order appear in the trading partner transactions whose IDs are previously given. After getting the table names, you can view the tables by selecting <i>Translation and Validation Tables</i> from the Administrator's Menu (MP01).

The Invoice Layout

Figure 14-2 on page 14-19 is an example of an invoice as it might be displayed using the predefined document layout. To view or print a document whose application data format has been mapped to a layout, select *Business documents* from the Interactive Entry Facility Menu (MM01). Then select *View* or *Print* for any document that uses that application data format. To create a formatted document in a file, select *File* and then *File formatted*.

The following are associated with the predefined invoice:

Item	Description
Data format IDs	GENINV (for sending) and GENINVRCV (for receiving). You can view or print these application data formats after selecting <i>Application Data Formats</i> from the Administrator's Menu (MP01).
Transaction IDs	GENINVMAP (for sending) and GENINVRCVMAP (for receiving). You can print the definitions of these transactions after selecting <i>Trading Partner Transactions</i> from the Administrator's Menu (MP01).
Tables	The names of the tables used in mapping the invoice appear in the trading partner transactions whose IDs are given previously. After getting the table names, you can view the tables by selecting <i>Translation and Validation Tables</i> from the Administrator's Menu (MP01).

Document Layout Messages

The Document Layout Utility provides messages for the following:

- File errors
- DataInterchange services errors
- Layout statement errors

Correct file errors and statement errors using the information in this section and the descriptions of the layout statements. Report DataInterchange errors to support personnel.

When the utility finds an error in a layout statement, it prints a message along with the input line number of the statement and the message ID. Use the message ID to refer to *DataInterchange Messages and Codes* for more information about the error. Certain error conditions affect the amount of checking performed on the remaining statements. Therefore, getting an error-free file of layout statements can take several attempts. A document layout is not added to the VSAM file until it has no errors.

Chapter 15. Invoking the DataInterchange Utility

Invoking the Utility from the Administrator's Menu	15-2
--	------

Chapter 15. Invoking the DataInterchange Utility

This chapter describes how you can invoke the DataInterchange Utility online in an MVS environment from either the Administrator's menu (MP01) or with a CLIST. The DataInterchange Utility provides command level access to DataInterchange services such as outbound translation processing, inbound translation processing, customizing, and reporting.

Note: Inbound and outbound translation processing may also be requested directly from an application program using the DataInterchange Application Program Interface (API). More information on the DataInterchange API can be found in the *DataInterchange Programmer's Reference*.

DataInterchange provides various methods to invoke the utility. This chapter focuses on options 1 and 2. Information for the remaining methods are documented in the *DataInterchange Programmer's Reference*.

1. Main menu option 12 - Utility (TSO only)

Select this method when you want to invoke the utility to test your mapping and trading partner setup using the DataInterchange panels.

2. TSO CLISTs EDIUVSM or EDIADB2 (TSO only)

Select this method when you want to invoke the utility to test your mapping and trading partner setup. This method should be used for testing purposes. Sample CLISTs can be found in the data set EDI.VvRMM.SEDISAM1 where VvRMM is Version, Release, and Modification level. For example, V2R1M0 is version 2, release 1, modification level 0.

3. JCL submitted to TSO batch

Select this method when your operations are mainly batch oriented. Your daily processing might use this method with either submitted JCL or a job scheduler. Sample JCL can be found in the data set EDI.VvRMM.SEDISAM1 where VvRMM is Version, Release, and Modification level. For example, V2R1M0 is version 2, release 1, modification level 0.

4. API call to the DataInterchange Utility Service (CICS ONLY)

Select this method when your operations are mainly real-time oriented. The utility initializes the processing environment, invoke the EDI processing, and terminates the processing. Your daily processing might use this method.

5. API call to the DataInterchange Utility Service HOT-DI mode (CICS ONLY)

Select this method when your operations are mainly real-time oriented. The DataInterchange API initializes the processing environment, the utility executes the EDI processing, and the DataInterchange API terminates the processing environment. This allows initialization and termination control of the DataInterchange processing environment. Your daily processing might use this method.

6. CICS transaction EDIW (CICS ONLY)

Select this method when your operations are mainly real-time oriented and you want to test your mapping and trading partner setup. This method should be used for testing purposes.

The Administrator's menu option allows you to allocate special files, enter utility (PERFORM) commands, and invoke the utility while running DataInterchange administration.

The CLIST method allows you to allocate special files, enter perform statements, and invoke the utility in a stand-alone, online environment independent of DataInterchange administration.

Invoking the Utility from the Administrator's Menu

To invoke the utility from the Administrator's Menu, follow these steps:

1. Select *Utility* from the Administrator's Menu (MP01). The DataInterchange for MVS Utility Invocation panel (FF11) is displayed.

FF11		DataInterchange for MVS Utility Invocation	
Command file name	_____	
Print file name	_____	
Report file name	_____	
Exception file name	_____	
Tracking file name	_____	
Query file name	_____	
Work file name	_____	
Command delimiter	_____	
Pre-execution TSO command	..	_____	
Post-execution TSO command	..	_____	

2. Complete the following fields:

Note: You can optionally place the name of an MQSeries Queue in the following fields: Print file name, Report file name, Exception file name, and Tracking file name.

In this field:

Enter:

Command file name

The fully-qualified name of the sequential file that contains the utility command (perform) statements you want to use to run the utility. This field is optional, and if entered, the file is read and displayed on the Utility Command Statements panel (FF12). If the name is omitted, the panel is displayed with blank lines.

Print file name

The fully-qualified name of the sequential file to contain the report summarizing the results of the utility execution. This field is optional. The default is the file allocated to PRTFILE. Optionally, this file can be an MQSeries Queue.

Report file name

The fully-qualified name of the sequential file to contain the results of a transaction store report. This field is optional. The default is the file allocated to RPTFILE. Optionally, this file can be an MQSeries Queue.

Exception file name

The fully-qualified name of the sequential file to contain translated transactions that cannot be stored in the application file, for example, if the application file cannot be opened. This field is optional. The default is the file allocated to FFSEXCP. Optionally, this file can be an MQSeries Queue.

Tracking file name

The fully-qualified name of the sequential file to contain the optional records that result from a translation. This field is optional. The default is the file allocated to FFSTRAK. Optionally, this file can be an MQSeries Queue.

Query file name

The fully-qualified name of the sequential file to contain the results of a QUERY or DATA EXTRACT. This field is optional. The default is the file allocated to EDIQUERY.

In this field:	Enter:
Work file name	The fully-qualified name of the sequential work file that DataInterchange uses during send translation. This field is optional. The default is the file allocated to FFSWORK.
Command delimiter	The character to be used in place of the left or right parenthesis to enclose values in the DataInterchange Utility command language. This field is optional but is required if any of the values in the perform statements contain either a left or right parenthesis.
Pre-execution TSO command	<p>The TSO command, if any, to be executed prior to calling the utility. You can use this to specify a CLIST to be executed to allocate special files prior to running the utility. This field is optional.</p> <p>Note: Do not to change the normal allocation of files that takes place when you log on as the DataInterchange Administrator.</p>
Post-execution TSO command	The TSO command, if any, to be executed after completion of the utility. You can use this to specify a CLIST to be executed to free special files after the utility completes. This field is optional.

Note: Do not change the normal allocation of files that takes place when you log on as the DataInterchange Administrator.

3. Press Enter. The Utility Command Statements panel (FF12) is displayed.

[illegible]

If you specified a command file name on the previous panel (FF11), the first 79 characters of the first 100 records of that file are displayed.

You can do one of the following:

- Change any of the displayed command (perform) statements
- Enter new command (perform) statements
- Enter an asterisk (*) in position 1 to prevent the command from being executed by the utility

4. Press Enter. The utility service is called, and upon completion, the Utility Command Statements panel (FF12) is redisplayed with a message indicating either a successful completion or an unsuccessful completion with a completion code.

5. Press F3 (Exit) to return to the Administrator's Menu (MP01) or press F12 (Cancel) to return to the DataInterchange for MVS Utility Invocation panel (FF11).

Appendixes

Appendix A. Security	A-1
RACF Class for DataInterchange	A-1
The Resource Names	A-1
Controlling Access to Menu Options	A-2
Profile Data	A-5
Event Logging	A-5
Trading Partner Transactions	A-6
EDI Standards	A-6
Application Data Formats	A-6
Translation and Validation Tables	A-6
Envelope Standards	A-7
Translation and Communication Services	A-7
Export/Import Utility	A-8
Migrating a Transaction Mapping	A-8
Archiving/Restoring Event Log Records and Removing Store Documents	A-8
Where Resource Names Are Checked	A-8
Using RACF General Resources Services	A-13
 Appendix B. Using the 841 Transaction Set	 B-1
The BIN Segment ID	B-1
Length of the BIN Segment	B-1
The EFI Segment	B-1
Send Processing for the Binary Segment	B-2
Receive Processing for the Binary Segment	B-7
 Appendix C. The IBM DataInterchange Web Site	 C-1

Appendix A. Security

Security for DataInterchange is provided by the Resource Access Control Facility (RACF**) or an equivalent product that is consistent with System Authorization Facility (SAF) interfaces. To protect DataInterchange programs and data, use RACF (or an equivalent) and the resource names described later in this appendix.

DataInterchange provides the level of security that RACF or an equivalent can provide in the MVS/TSO environment. This lets you control access to the DataInterchange systems, menu options, and files during DataInterchange execution. DataInterchange provides control over the record-level access through predefined resource names. However, because RACF or an equivalent provides security only to the data set level, users have access to the entire data set.

If you have installed more than one DataInterchange system, have multiple system-related resource names, and have a user that needs to have different authority levels based on the system-related resource names, the only way to guarantee the authority levels is to use a different MVS/TSO user ID for each system-related resource name.

RACF Class for DataInterchange

To protect DataInterchange resources, create a new RACF class called EDIR. Add this class to the class descriptor table (CDT) using the ICHERCDE macro. For details, see your access control facility documentation. For EDIR, specify the macro as follows. Values not shown are chosen locally.

```
ICHERCDE CLASS=EDIR,
  MAXLNTH=39,
  FIRST=ANY,
  OTHER=ANY
```

You must also add class EDIR to the RACF router table using the ICHRFRTB macro with ACTION = RACF:

```
ICHRFRTB CLASS=EDIR,
  ACTION=RACF
```

Customers using TopSecret instead of RACF should add a RESCLASS, EDIR, to their Resource Descriptor Table (RDT) and manage it as any resource.

The Resource Names

The following resource names are RACF profile names. Each resource name consists of two or more qualifiers connected by periods. Qualifiers that appear in uppercase are to be used as shown. Those in lowercase are variables. The variable `sys`, for system name, is required only if you have installed more than one copy of DataInterchange on the same system. For example, you would use `sys` to distinguish between a French and a German version, or between a test and a production version.

You can use two types of system-related resource names:

- `SYSTEM.sys`, where `sys` is the system name for one copy of DataInterchange, such as the copy used for testing or the copy used for production. Using system names, you can provide separate protection for each copy of DataInterchange. If using a CLIST to start DataInterchange, the CLIST must define

the system name, which can be up to 8 characters. If a system name is not provided when DataInterchange is started, DIENU is used by default.

- SYSTEM, which determines how the system name is used to grant users access to the DataInterchange product.
 - If the SYSTEM resource is defined, users must be specifically granted access under the SYSTEM.sys resource name, or they are denied access to the product.
 - If the SYSTEM resource is not defined, users are granted access to the product unless they are specifically excluded under the SYSTEM.sys resource name.
 - For all resource names which have variables at the end of the name, you can customize your access to the resource.

Defining Levels of Access

You can grant the following access levels to resource names:

Level	Description
None	Users cannot access the resource.
Read	Allows users to view the resource.
Update	Allows users to view and update the resource. It does not allow the user to copy or create the resource.
Alter	Allows users to view, update, create, copy, or delete the resource.

Controlling Access to Menu Options

You can control access to the options on the following DataInterchange menus:

- Administrator's Menu (MP01)
- Interactive Entry Facility menu (MM01)
- Transaction Store Facility menu (TF01)

Administrator's Menu (MP01)

Use the following resource names to control user access to the options on the Administrator's Menu (MP01). Only users who have access under these resource names see the associated options on their menu. If a user is not authorized to a resource name, a violation will be issued.

Note: Turning off auditing for menu options is recommended because these violations are normal.

Resource Name	Controls Access to:	Access Level
sys.MENU.IEF	<i>Interactive Entry Facility Menu (MM01)</i>	In addition, users must have access to one of the Interactive Entry Facility Menu (MM01) options (see "Interactive Entry Facility Menu (MM01)" on page A-4). The level of access is not important for this option.
sys.MENU.PROFILE	<i>Profiles option</i>	The level of access is not important for this option.
sys.MENU.EVENT	<i>Event logging option</i>	The level of access is not important for this option. Access to event logging is further restricted by the resource name <i>sys.EVENT.userid</i> .
sys.MENU.TRANSACTION	<i>Trading partner transactions option</i>	Only users with ALTER access to the resource can create new trading partner transactions.
sys.MENU.STANDARD	<i>EDI standards option</i>	Only users with ALTER access to the resource can create new standards.
sys.MENU.FORMAT	<i>Application data formats option</i>	Only users with ALTER access to the resource can create new application data formats.
sys.MENU.TABLE	<i>Translation and validation tables option</i>	Only users with ALTER access to the resource can create new tables.
sys.MENU.ENVELOPE	<i>Envelope standards option</i>	Only users with ALTER access to the resource can create new envelope standards.
sys.MENU.TSF	<i>Transaction Store Facility Menu (TF01)</i>	In addition, users must have access to one of the Transaction Store Facility Menu (TF01) options (see "Transaction Store Facility Menu (TF01)" on page A-4.) The level of access is not important for this option.
sys.MENU.EXPORT	<i>Export Menu (EI00)</i>	The level of access is not important for this option. There are no additional restrictions for this option.
sys.MENU.IMPORT	<i>Import Menu (EI01)</i>	The level of access is not important for this option. There are no additional restrictions for this option.
sys.MENU.UTILITY	<i>Utility option</i>	The level of access is not important for this option.

Interactive Entry Facility Menu (MM01)

Use the following resource names to control access to the options on the Interactive Entry Facility Menu (MM01). Only users who have access under these resource names see the associated options on their menu. The level of access is not important for these options.

Resource Name	Controls Access to:
sys.IEF.DOCUMENT	Business documents option
sys.IEF.MESSAGES	Messages option
sys.IEF.FILES	File requests option
sys.IEF.DEFINITION	Business document definitions option

Transaction Store Facility Menu (TF01)

Use the following resource names to control access to the options on the Transaction Store Facility Menu (TF01). Only users who have access under these resource names see the associated options on their menu. The level of access is not important for these options.

Resource Name	Controls Access to
sys.TSF.ENVELOPE	Envelope transactions and Reenvelope transactions options
sys.TSF.SEND	Envelope and send transactions, and Reenvelope and send transactions options
sys.TSF.RECEIVE	Receive and deenvelope transactions option
sys.TSF.TRANSLATE	Translate received transactions and Re-translate received transactions options
sys.TSF.REPORT	Report transaction status and Report interchange/group options
sys.TSF.UPDATE	Update store status and Update interchange/group status options
sys.TSF.NETSTAT	Update network status option

Profile Data

All the DataInterchange profiles are protected with the following resource name:

sys.PROF.profname.mbrname

where:

mbrname

is the name of a profile member

profname

is one of the following:

Profile	Description
ACTLOGS	Activity logs
ADAMCTL	Exit routine definitions
APPDEFS	Application definitions
CONTRECV	Continuous receive
E	EDIFACT standard envelope data
I	ICS standard envelope data
LANGPROF	Language profile
MQSERIES	MQSeries queue profile
NETOP	Network operations
NETPROF	Networks
REQPROF	Requestors
SECUPROF	Encryption, authentication, and filtering definitions
SYSPROF	Systems
T	UNTDI standard envelope data
TPPROF	Trading partners
U	UCS standard envelope data
X	X12 standard envelope data

Only users with ALTER access to the resource *sys.PROF.profname* can create or copy profile members within that profile. For example, ALTER authority on *sys.PROF.TPPROF* allows the user to add and copy trading partner profile members.

Only users with ALTER access authority to the resource *sys.PROF.profname* can import profile members into that profile. The user must also have ALTER access authority to resource *sys.PROF.profname.mbrname* for any members that are to be imported. This also applies to profile members imported as associated objects.

Users with ALTER access to the resource *sys.PROF.** can create or copy profile members in **any** profile unless specifically excluded under the resource name for the specific profile.

Event Logging

sys.MENU.EVENT controls access to the Event logging option. Access to event log records is further restricted by the resource name *sys.EVENT.userid*.

Trading Partner Transactions

Trading partner transactions are protected under the resource name `sys.TRANSACTION.tptname`. Only users with ALTER access to the resource name `sys.MENU.TRANSACTION` can create or copy trading partner transactions.

Note: A trading partner transaction refers to many other resources within DataInterchange, including:

- Standards
- Application data formats
- Profiles
- Tables

A user with access to a trading partner transaction has **implied** access to all resources referred to by that transaction. The programs, except for export and import, do not check for explicit access to each resource referred to by the transaction definition.

The user must have ALTER access authority to the resource `sys.TRANSACTION.tptname` to import that specific trading partner transaction. Users must have UPDATE access to generate control strings. If the user has any access to the resource `sys.TRANSACTION.tptname`, then the user will be able to issue the MAP command and change the mapping.

EDI Standards

EDI standards are protected under the resource name `sys.STANDARD.stdname`. Only users with ALTER access to the resource name `sys.MENU.STANDARD` can create or copy EDI standards.

The user must have ALTER access authority to the resource `sys.STANDARD.stdname` in order to import that specific EDI standard. This also applies to EDI standards imported as associated objects.

Application Data Formats

Application data formats are protected under the resource name `sys.FORMAT.fmtname`. Only users with ALTER access to the resource name `sys.MENU.FORMAT` can create or copy application data formats.

The user must have ALTER access authority to the resource `sys.FORMAT.fmtname` to import that specific application data format. This also applies to application data formats imported as associated objects.

In order to create a standard from an application data format, the user must have read READ access to `sys.FORMAT.fmtname` and ALTER access to `sys.MENU.STANDARDS`. Additionally, if the standard already exists, the user must have ALTER access to `sys.STANDARD.stdname`.

Translation and Validation Tables

Translation and validation tables are protected under the resource name `sys.TABLE.tblname`. Only users with ALTER access to the resource name `sys.MENU.TABLE` can create or copy tables.

The user must have ALTER access authority to the resource `sys.TABLE.tblname` to import that specific table. Tables are imported as associated objects only.

Envelope Standards

Envelope standards are protected under the resource name `sys.ENVELOPE.stdname`. Only users with ALTER access to the resource name `sys.MENU.ENVELOPE` can create or copy envelope standards.

You can customize access to this resource so that a user cannot add items to the database if they begin with a character string you specify. The user must have ALTER access authority to the resource `sys.ENVELOPE.stdname` to import that specific envelope standard. This also applies to envelope standards imported as associated objects.

Translation and Communication Services

DataInterchange provides services (function calls) that are available through an application program interface (API). The Interactive Entry Facility, Transaction Store Facility, and DataInterchange Utility also use this interface. The resource names used to protect these functions are the following:

Resource Name	Controls Access to:	Access Level
<code>sys.FUNCTION.TRANSLATE.SEND</code>	<i>Translate for Sending</i> function	Only users who have access under this resource can translate documents for sending. In addition, users must have access under the appropriate <code>sys.TRANSACTION.tptname</code> resource, or the translation is not allowed.
<code>sys.FUNCTION.TRANSLATE.RECEIVE</code>	<i>Translate Received</i> <i>Transactions</i> function	Only users who have access under this resource can translate documents that have been received. In addition, users must have access under the appropriate <code>sys.TRANSACTION.tptname</code> resource, or the translation is not allowed.
<code>sys.FUNCTION.SEND</code>	<i>Send Network</i> function	Only users who have access under this resource can send documents to the network. In addition, users must have access under the appropriate <code>sys.PROF.REQPROF.mbrname</code> and <code>sys.PROF.TPPROF.mbrname</code> , or the send is not allowed.
<code>sys.FUNCTION.ENVELOPE</code>	<i>Envelope</i> and <i>Deenvelope</i> functions	Only users who have access under this resource name can envelope or deenvelope documents.
<code>sys.FUNCTION.CANCEL</code>	<i>Network Cancel</i> function	Only users who have access under this resource can cancel documents from the network or use the RECALL function from the Interactive Entry Facility. In addition, users must have access under the appropriate <code>sys.PROF.REQPROF.mbrname</code> and <code>sys.PROF.TPPROF.mbrname</code> , or the cancel is not allowed.
<code>sys.FUNCTION.RECEIVE</code>	<i>Receive Network</i> function	Only users who have access under this resource can receive documents from the network. In addition, users must have access under the appropriate <code>sys.PROF.REQPROF.mbrname</code> and <code>sys.PROF.TPPROF.mbrname</code> , or the receive is not allowed.

Export/Import Utility

The Export/Import Utility is protected under the resource names `sys.FUNCTION.EXPORT` and `sys.FUNCTION.IMPORT`. Only users with access under these names can run the utility. The Export/Import Utility checks your authority to individual resource names of objects before it performs the export or import. To export an object, you must have READ access to that object. To import an object, you must have ALTER access to that object.

Migrating a Transaction Mapping

This function is protected under the resource name `sys.FUNCTION.MIGMAP`. Only users with ALTER access can use this function.

Archiving/Restoring Event Log Records and Removing Store Documents

These functions are protected under the resource name `sys.FUNCTION.EVENT.ARCHIVE`. Only users with access under this name can request archiving of event logs or removing of documents from the Transaction Store by the REMOVE TRANSACTIONS command.

The JCL statements produced by the Archive action use DIENU for the system name parameter. If you defined some other system name, correct the name in the JCLOUT file before running the archive job.

Where Resource Names Are Checked

Table A-1 lists the DataInterchange resource names and the services that check the user's authorization.

Table A-1 (Page 1 of 4). Where Resource Names are Checked

Resource Name	Checked by
SYSTEM SYSTEM.sys	DataInterchange during initialization
sys.MENU.TRANSACTION sys.MENU.STANDARD sys.MENU.ENVELOPE sys.MENU.FORMAT sys.MENU.TABLE sys.MENU.IEF sys.MENU.TSF sys.MENU.UTILITY	Menu Processor during startup to determine which menu options are displayed to the current user
sys.MENU.TRANSACTION sys.TRANSACTION. <i>tpname</i>	Trading Partner Transactions before a transaction mapping is copied or added
sys.TRANSACTION. <i>tpname</i>	<ul style="list-style-type: none"> Trading Partner Transactions before giving access to <i>tpname</i> Interactive Entry Facility before sending a document Translator before a send or receive action against <i>tpname</i> Export before exporting <i>tpname</i> Import before importing <i>tpname</i> Migration Mapping before migrating <i>tpname</i>

Table A-1 (Page 2 of 4). Where Resource Names are Checked

Resource Name	Checked by
sys.MENU.STANDARD sys.STANDARD. <i>stdname</i>	Standards Customization before an EDI standard is copied or added
sys.STANDARD. <i>stdname</i>	<ul style="list-style-type: none"> Standards Customization before giving access to <i>stdname</i> Export before exporting <i>stdname</i> Import before importing <i>stdname</i> Migration Mapping before using <i>stdname</i>
sys.MENU.ENVELOPE sys.ENVELOPE. <i>envlname</i>	Envelope Standards before an envelope standard is copied or added
sys.ENVELOPE. <i>envlname</i>	<ul style="list-style-type: none"> Envelope Standards before giving access to <i>envlname</i> Export before exporting <i>envlname</i> Import before importing <i>envlname</i>
sys.MENU.FORMAT sys.FORMAT. <i>fmtname</i>	Application Data Formats before an application data format is copied or added
sys.FORMAT. <i>fmtname</i>	<ul style="list-style-type: none"> Application Data Formats before giving access to <i>fmtname</i> Interactive Entry Facility before: <ul style="list-style-type: none"> Application data format selections are listed Documents are listed Adding, copying, updating, viewing, filing, or printing from the list panel command line Restoring a document Interactive Entry Facility to restrict access to <i>fmtname</i> when listing document definitions and before adding a document definition Transaction Store Facility when the application data format ID is entered as a selection criterion Transaction Store Service to return only authorized transactions from the Transaction Store database Export before exporting <i>fmtname</i> Import before importing <i>fmtname</i>
sys.MENU.TABLE	Table Maintenance before copying or adding a table
sys.TABLE. <i>tblname</i>	<ul style="list-style-type: none"> Table Maintenance before giving access to <i>tblname</i> Export before exporting <i>tblname</i> Import before importing <i>tblname</i>
sys.PROF. <i>profname</i> sys.PROF. <i>profname.mbrname</i>	Profile Maintenance before copying or adding a member in <i>profname</i>
sys.PROF. <i>profname</i>	<ul style="list-style-type: none"> Export before exporting members from <i>profname</i> Import before importing members into <i>profname</i>
sys.PROF. <i>profname.mbrname</i>	<ul style="list-style-type: none"> Profile Maintenance and Profile Services before access is given to <i>profname.mbrname</i> Export before exporting member <i>mbrname</i> from <i>profname</i> Import before importing member <i>mbrname</i> into <i>profname</i>

Table A-1 (Page 3 of 4). Where Resource Names are Checked

Resource Name	Checked by
<i>sys.PROF.TPPROF.mbrname</i>	<ul style="list-style-type: none"> • Communications before actions involving <i>mbrname</i> • Interactive Entry Facility before: <ul style="list-style-type: none"> – Displaying the list of documents – Adding, copying, deleting, updating, viewing, filing, printing, sending, or translating from the document list panel command line – Restoring a document • Interactive Entry Facility to give access to message and file requests only where access to <i>mbrname</i> is granted • Transaction Store Facility when the trading partner nickname is entered as a selection criterion or as a receive parameter • Transaction Store Service to return only authorized transactions from the Transaction Store database
<i>sys.PROF.REQPROF.mbrname</i>	<ul style="list-style-type: none"> • Communications before actions involving <i>mbrname</i> • Interactive Entry Facility upon entry of <i>mbrname</i> before sending or receiving of documents, messages, and files • Transaction Store Facility when the requestor ID is entered as a send or receive parameter
<i>sys.EVENT.userid</i>	Event Logging before displaying a log record to the current user
<i>sys.FUNCTION.TRANSLATE.SEND</i>	<ul style="list-style-type: none"> • Translator before a send translate function • Interactive Entry Facility before sending or translating documents
<i>sys.FUNCTION.TRANSLATE.RECEIVE</i>	<ul style="list-style-type: none"> • Translator before a receive translate function • Interactive Entry Facility before receiving documents, messages, or files • Transaction Store Facility to determine whether <i>Translate</i> and <i>Re-translate</i> should appear on the Transaction Store Facility Menu (TF01)
<i>sys.FUNCTION.MIGMAP</i>	Migration Mapping before migrating a transaction mapping
<i>sys.FUNCTION.ENVELOPE</i>	<ul style="list-style-type: none"> • Translator before an envelope or develope function • Transaction Store Facility to determine whether <i>Envelope</i>, <i>Reenvelope</i>, <i>Envelope and Send</i>, and <i>Reenvelope and Send</i> should appear on the Transaction Store Facility Menu (TF01)
<i>sys.FUNCTION.SEND</i>	<ul style="list-style-type: none"> • Communications before a send function • Interactive Entry Facility before sending documents, messages, or files • Transaction Store Facility to determine whether <i>Envelope and Send</i> and <i>Send</i> should appear on the Transaction Store Facility Menu (TF01)
<i>sys.FUNCTION.RECEIVE</i>	<ul style="list-style-type: none"> • Communications before a receive function • Interactive Entry Facility before receiving documents, messages, or files • Transaction Store Facility to determine whether <i>Receive</i> should be displayed on the Transaction Store Facility Menu (TF01)

Table A-1 (Page 4 of 4). Where Resource Names are Checked

Resource Name	Checked by
sys.FUNCTION.CANCEL	<ul style="list-style-type: none"> Communications before a cancel function Interactive Entry Facility before recalling documents, messages, or files
sys.FUNCTION.EXPORT	Export Utility before a batch export
sys.FUNCTION.IMPORT	Import Utility before a batch import
sys.MENU.IEF sys.IEF.DOCUMENT sys.IEF.MESSAGES sys.IEF.FILES sys.IEF.DEFINITION	Menu Processor to determine which options on the Interactive Entry Facility Menu (MM01) should be displayed to the current user and whether to display the Interactive Entry Facility option on the Administrator's Menu (MP01)
sys.MENU.TSF sys.TSF.TRANSLATE sys.TSF.ENVELOPE sys.TSF.SEND sys.TSF.RECEIVE sys.TSF.UPDATE sys.TSF.REPORT sys.TSF.NETSTAT	Menu Processor to determine which options on the Transaction Store Facility Menu (TF01) should be displayed to the current user and whether to display the Transaction Store Facility option on the Administrator's Menu (MP01)

Security Examples

The following examples show the resource names (RACF profiles) with which you associate a user's ID to enable the user to perform certain tasks.

Example 1: Authorize an IEF user to enter and send purchase orders to trading partner ABC. Only one copy of DataInterchange is installed.

Required Resource	For Access to:
DIENU.MENU.IEF	Interactive Entry Facility option on the Administrator's Menu (MP01)
DIENU.IEF.DOCUMENT	Business documents option on the Interactive Entry Facility Menu (MM01)
DIENU.PROF.TPPROF.ABC	Trading partner ABC (required for data entry)
DIENU.FORMAT.SENDPO	Purchase order application data format SENDPO (required for data entry)
DIENU.TRANSACTION.ABCPO	Purchase order transaction ABCPO (required for sending)
DIENU.PROF.REQPROF.HENDER	Requestor profile member HENDER (required for sending)
DIENU.FUNCTION.TRANSLATE.SEND	Translate for sending function (required for sending)
DIENU.FUNCTION.SEND	Send function (required for sending)

Example 2: Authorize an EDI administrator to add trading partners. Only one copy of DataInterchange is installed.

Security

Required Resource

DIENU.MENU.PROFILE

DIENU.PROF.TPPROF.*

DIENU.FORMAT.*

For Access to:

Profiles option on the Administrator's Menu (MP01)

All trading partner profiles

All application data formats

Example 3: Authorize an EDI administrator to print status reports for transaction activity associated with trading partner ABC and data format SENDPO. Two copies of DataInterchange are installed, TEST and LIVE. Authorize this user for the copy with the system name TEST. Deny this user access to the LIVE system.

Required Resource

SYSTEM.TEST

For Access to:

Test system

Note: If the SYSTEM resource is defined, user access to SYSTEM.TEST or SYSTEM.LIVE must be explicit. If SYSTEM is not defined, user access to SYSTEM.LIVE must be explicitly denied.

TEST.MENU.TSF

Status reporting option on the Administrator's Menu (MP01)

TEST.TSF.REPORT

Status reporting option on the Transaction Store Facility Menu (TF01)

TEST.PROF.TPPROF.ABC

Trading partner ABC

TEST.FORMAT.SENDPO

Application data format SENDPO

Example 4: Authorize an EDI administrator to add new standards except for X12. Only one copy of DataInterchange is installed.

Required Resource

DIENU.MENU.STANDARD

Purpose

For access to the EDI standards option on the Administrator's Menu (MP01). Use ALTER access for this resource name.

DIENU.STANDARD.X12*

Limit access to all standards beginning with the string "X12" to READ access only.

Using RACF General Resources Services

The following steps show an example of defining a RACF profile for the general resource SYSTEM.DIENU and then granting access to the resource.

1. From the RACF - Services Option Menu, select option 2 to add a profile for a general resource.

```

OPTION ==> 2          RACF - SERVICES OPTION MENU

SELECT ONE OF THE FOLLOWING:

1  DATA SET          ADD, CHANGE, DELETE, or DISPLAY the profile
                        for a data set.
2  GENERAL RESOURCE   ADD, CHANGE, DELETE, or DISPLAY the profile
                        for a general resource.
3  GROUP              ADD, CHANGE, DELETE, or DISPLAY a group profile.
                        CONNECT or REMOVE users.
4  USER              ADD, CHANGE, DELETE, or DISPLAY a user profile.
                        Change a user's password.
5  SYSTEM OPTIONS     DISPLAY or SET the system wide security options.
                        REFRESH in-storage profile lists.
T  TUTORIAL           View a general description of RACF.
  
```

The RACF - General Resource Services panel is displayed.

```

OPTION ==> 1          RACF - GENERAL RESOURCE SERVICES

SELECT ONE OF THE FOLLOWING:

1  ADD      Add a profile          D  DISPLAY  Display profile contents
2  CHANGE   Change a profile       S  SEARCH   Search RACF data set for
                                     profiles
3  DELETE   Delete a profile
4  ACCESS   Maintain access list
5  AUDIT    Monitor access attempts
             (for auditors only)

ENTER RESOURCE PROFILE INFORMATION

RESOURCE CLASS   ==> edir
RESOURCE NAME    ==> system.dienu

USE MODEL PROFILE ==>          YES if the profile is to be modeled
  
```

2. Complete the fields as follows:
 - a. Type **1** in the *Option* field to add a profile.
 - b. Type **EDIR** in the *RESOURCE CLASS* field.
 - c. Type **SYSTEM.DIENU** in the *RESOURCE NAME* field.
 - d. Press Enter.

The RACF - Add General Resource Profile panel is displayed.

```

                                RACF - ADD GENERAL RESOURCE PROFILE
COMMAND ==>

    CLASS: EDIR    PROFILE NAME: SYSTEM.DIENU

ENTER OR CHANGE RESOURCE PROFILE INFORMATION
OWNER          ==>  USERA      Userid or group name
LEVEL          ==>  0          0-99
FAILED ACCESSES ==>  FAIL      FAIL or WARN
UACC           ==>  NONE      NONE, READ,  UPDATE,  CONTROL, or ALTER
AUDIT SUCCESSES ==>  NOAUDIT  READ, UPDATE, CONTROL, ALTER, or NOAUDIT
AUDIT FAILURES  ==>  READ     READ, UPDATE, CONTROL, ALTER, or NOAUDIT
NOTIFY         ==>           Userid

TO ADD OPTIONAL INFORMATION, ENTER YES:
INSTALLATION DATA ==>
APPLICATION DATA  ==>
TAPE VOLUME INFORMATION ==>      TAPEVOL class only
GROUP MEMBERS      ==>          Resource group classes only
ACCESS LIST        ==>
TERMINAL INFORMATION ==>      TERMINAL class only
SECURITY LEVEL/CATEGORIES ==>

```

3. Press Enter to accept the information supplied on the Add panel.

The RACF - General Resources Services panel is redisplayed.

4. Select option 4 to grant users access to the resource you just defined.

```

                                RACF - GENERAL RESOURCE SERVICES
OPTION ==> 4

SELECT ONE OF THE FOLLOWING:

    1 ADD      Add a profile          D DISPLAY  Display profile contents
    2 CHANGE   Change a profile       S SEARCH   Search RACF data set for
                                           profiles
    3 DELETE   Delete a profile
    4 ACCESS   Maintain access list
    5 AUDIT    Monitor access attempts
               (for auditors only)

ENTER RESOURCE PROFILE INFORMATION

RESOURCE CLASS  ==>  edir
RESOURCE NAME   ==>  system.dienu

USE MODEL PROFILE ==>          YES if the profile is to be modeled

```

The RACF - Maintain General Resource Access List - Add panel is displayed.

5. Grant ALTER access rights for the resource SYSTEM.DIENU. The owner, USERA, always has access. Enter the IDs of all other users you want to grant ALTER access to this resource.

```

RACF - MAINTAIN GENERAL RESOURCE ACCESS LIST - ADD
COMMAND ==>

CLASS: EDIR   PROFILE NAME: SYSTEM.DIENU

ENTER AUTHORITY TO BE GRANTED:
ACCESS AUTHORITY ==> alter  NONE, READ, UPDATE, CONTROL, or ALT

ENTER USER/GROUP ID TO BE ADDED:
==> usera ==> ==> ==>
==> userb ==> ==> ==>
==> userc ==> ==> ==>
==>      ==> ==> ==>
==>      ==> ==> ==>
==>      ==> ==> ==>
==>      ==> ==> ==>

ENTER INFORMATION FOR PROFILE TO BE COPIED:
PROFILE NAME ==>
CLASS        ==> EDIR
GENERIC      ==> yes      YES If the profile name is generic
VOLUME SERIAL ==>      If a non-cataloged data set profile

```

Appendix B. Using the 841 Transaction Set

The ASC-X12 Specifications/Technical Information transaction set (841) defines a way for trading partners to exchange technical information the same way they exchange EDI transactions. This technical information, which can be graphic, image, or audio, can contain binary data. The binary data can assume any value in the range hex 00 through hex FF.

In the syntax of X12 transaction sets, data elements that are separated by delimiters are combined into a segment that is identified by a segment ID and terminated with a segment delimiter. The binary data introduced by the 841 transaction set causes problems for this syntax because the binary data may contain a value that matches a segment delimiter. Translators and networks that support the 841 transaction set must have a way to identify binary data and determine its length so that it does not interfere with parsing the rest of the envelope. Special care must be taken if you want to send and receive files between DataInterchange and translators on other platforms. Not all operating systems support the record types MVS uses. For more information, see "Format Specifications" on page B-3.

The BIN Segment ID

The binary data is identified with a BIN segment ID, which notifies the parser that data following the segment ID is binary. Although the parser must always treat the BIN segment as if it contained binary data, the segment can contain normal text. The first data element of the BIN segment contains the length of the binary data so that the parser knows the amount of data to pass without interference. The first character after the binary data should be BIN segment terminator. Any other value is a syntax error that ends parsing for the envelope.

The BIN segment ID triggers the special binary processing. Although the 841 transaction set is the only one that uses the BIN segment, binary processing is not limited to the X12 standard. In addition, DataInterchange applies this special processing to all envelope types.

Length of the BIN Segment

The value in the length data element of the BIN segment can be up to 15 characters long, which means the maximum length of a BIN segment is 999,999,999,999 bytes (fifteen 9s). However, the maximum size that DataInterchange supports is 2,147,483,647 bytes, which is the maximum signed integer value that a fullword can hold. The binary segment is a repeating segment with an unlimited number of repetitions, so as far as the standard is concerned, it is unlimited.

The EFI Segment

For the 841 transaction set, an Electronic Format Identification (EFI) segment precedes a group of repeating BIN segments. The EFI segment provides information about the binary data file. DataInterchange provides special processing for only the following data elements in the EFI segment:

- File name
- Block type
- Record length
- Block length

Data elements such as *security technique* and *compression technique* do not receive special processing. When data compression is used, the sender must compress the data before translation, and the receiver

Using the 841 Transaction Set

must decompress it after translation. Any security that occurs is for the entire transaction or for the group containing the transaction. DataInterchange does not provide a special security interface for data in the BIN segments.

Use of the four data elements listed above are covered later in the discussions of send and receive processing. The following information may help explain how DataInterchange treats two of them: file name and block type.

File name: The standard essentially defines file name as free format with a maximum length of 64 characters. Its format depends on the computer operating system being used. Accordingly, DataInterchange treats this element as a fully qualified data set name, including the owner ID. A file name value with a length greater than 44 characters (the maximum length of a data set name) will be ignored.

Block type: The standard defines block type as free format with a maximum length of 4 characters. Its value indicates the organization of data in the BIN segments. Examples are fixed length, variable length, and spanned. However, the definition does not provide any codes to represent the organizations, such as F for fixed and V for variable. In the absence of standard codes, DataInterchange interprets the block type as a string of characters that can have the following values:

Value	Description
A	ASA printer control characters
B	Block
F	Fixed
M	Machine code printer control characters
S	Spanned
U	Undefined
V	Variable

Send Processing for the Binary Segment

This section covers mapping a file or an application field to a binary segment.

Mapping Data from a File to a Binary Segment

The primary intent of the 841 transaction set and the BIN segment is to transmit files between trading partners. Therefore, DataInterchange provides a way for you to map a *file* to a data element in the standard. Normally, you would map a data element to an application field. Mapping data from a file to the BIN segment requires a new application data type, FN (file name).

The FN data type has special meaning only when it is mapped to the BIN02 data element of the BIN segment. At all other times, the FN data type is treated as an alphanumeric (AN) field. The special meaning for the FN data type when mapped to the BIN02 element is as follows:

- Rather than moving the field containing data to BIN02, DataInterchange moves the entire contents of a file named by the field to the BIN02 data element.
- DataInterchange automatically supplies the length (BIN01), based on the amount of data read from the file.

Note: The Interactive Entry Facility recognizes the FN data type, and enables you enter the name of a file that supplies data for a BIN segment. For IEF, the FN data type has the same characteristics as the CH data type.

Aside from the special processing described above, the mapping of an FN field is the same as any other field, including the use of a translation table or user exit routine. For example, the application may not

know the data set name of a file to be transmitted. The application may know it by a coded name. A translation table or a user exit provides a way of transforming the coded name to a data set name. You can also use a literal if the application data does not contain a value at all or if the value it supplies is all blanks. Because the name of the file mapped to the BIN segment can also be specified in the EFI segment, you should use the same field for mapping both segments.

If an EFI segment is built and it contains a value in the file name data element, that file name is used if the field mapped to the BIN02 data element is all blanks and no literal value is supplied.

The format of data in an FN type field is:

type:name

where:

type

is an optional value that indicates the type of file being provided. Valid values are:

Value	Description
DD	Data definition name (ddname)
DS	Data set name
MQ	MQSeries Queue profile member name
VS	VSAM entry sequenced data set
TD	Transient data queue
TM	Temporary storage queue (main)
TS	Temporary storage queue (auxiliary)

MQ, VS, TD, TM, and TS are for CICS only. The default is DS.

name

is either a ddname or data set name based on the value of *type*. It must conform to the conventions for its type. A ddname has a maximum length of 8. A data set name has a maximum length of 44.

name can also contain the literal &IV, which tells DataInterchange to substitute a value that represents the current binary file being processed. Each time a binary file is processed, DataInterchange increments this internal number. Thus, a specification of DD:EDIBF&IV equates to a ddname value of EDIBF1 the first time a binary file is processed, EDIBF2 the second time a binary file is processed, and so on.

The BIN segment, as mentioned earlier, can have a rather impressive length, and you may think that the entire file can be put into a single BIN segment. This is not always true. The number of BIN segments required to hold any file depends on the size of the file and the format of records in the file.

Format Specifications

A file should be transmitted in a form that permits the receiver to recreate an exact copy of the file.

For MVS based systems, a file consists of individual records and each record consists of some number of characters. Records may either have a fixed format where each record has the same number of characters, or a variable format where each record has a variable number of characters. For variable length records, the number of characters in the record is maintained in a record header which is the first 4 bytes of the record and has a format of LLXX, where LL is the number of characters in the record (including the length of the LLXX field) and XX is reserved.

Files on other systems (such as DOS or OS/2) do not have record boundaries but are treated as a stream of characters with record boundaries (if any) established by the program that is reading the stream. DataInterchange provides 8 different ways to combine data from a file into BIN segments so that the

Using the 841 Transaction Set

receiver of the file can recreate an exact copy of the file being sent. The 8 different ways represent combinations of how records are put into binary segments (either combined or individually) and the format of the record header within the BIN segment. There are four possible header formats for records within the BIN segment:

1. No header - the data is put into the binary without a header.
2. MVS format - each record in the binary segment is preceded by a record header of the form LLXX.
3. SHORT format - each record in the binary segment is preceded by a record header of the form LL.
4. LONG format - each record in the binary segment is preceded by a record header of the form LLLL.

DataInterchange, by default, transmits all files with a fixed format by combining the records into a single binary segment without any record headers. Because the file has a fixed format, the record headers are not needed for the receiving side to reconstruct the file.

DataInterchange, by default, transmits all files with a variable format by sending each record within the file as a single binary segment. Again record headers are not used because they are not necessary to be able to reconstruct the file.

If the defaults are not satisfactory, (for example, you want to send a file with variable length records as a stream of data in a single binary segment) then a special literal value can be supplied at mapping time that overrides the defaults. This literal is specified when mapping the binary element in the BIN segment (BIN02). You can either type it in the literal field on the mapping screen, or map an application field to the BIN02, which at run time supplies the literal to DataInterchange. This literal consists of a series of keywords shown below. Except for the first keyword (BinSpec), you can specify them in any order. In the descriptions below, the keywords are shown with acceptable abbreviations in CAPS:

Keyword	Description										
BinSpec	Signals that binary specifications follow.										
BINary(x)	Indicates how the records should be placed into a binary segment. Valid values for x are: <table><tr><th>Value</th><th>Description</th></tr><tr><td>C</td><td>Indicates records should be combined into a single binary segment (default for fixed record lengths)</td></tr><tr><td>R</td><td>Indicates each record should be a binary segment (default for variable record lengths).</td></tr></table>	Value	Description	C	Indicates records should be combined into a single binary segment (default for fixed record lengths)	R	Indicates each record should be a binary segment (default for variable record lengths).				
Value	Description										
C	Indicates records should be combined into a single binary segment (default for fixed record lengths)										
R	Indicates each record should be a binary segment (default for variable record lengths).										
Format(y)	Indicates the type of header that should precede each record in the binary segment. Valid values for y are: <table><tr><th>Value</th><th>Description</th></tr><tr><td>N</td><td>No header should be used (default).</td></tr><tr><td>V</td><td>LLXX header, where LL is the number of bytes in the record and XX is binary zeros.</td></tr><tr><td>S</td><td>LL header, where LL is the number of bytes in the record.</td></tr><tr><td>L</td><td>LLLL header, where LLLL is the number of bytes in the record.</td></tr></table>	Value	Description	N	No header should be used (default).	V	LLXX header, where LL is the number of bytes in the record and XX is binary zeros.	S	LL header, where LL is the number of bytes in the record.	L	LLLL header, where LLLL is the number of bytes in the record.
Value	Description										
N	No header should be used (default).										
V	LLXX header, where LL is the number of bytes in the record and XX is binary zeros.										
S	LL header, where LL is the number of bytes in the record.										
L	LLLL header, where LLLL is the number of bytes in the record.										
LIMIT(nnnnn)	If records are being combined, then <i>nnnnn</i> is the maximum size for a binary segment. After a binary segment reaches this limit, it is stopped and a new binary segment is started. The default limit is over 2 gigabytes or the amount of virtual storage available to the program, whichever is lower.										

Examples

Sending PC Executable Files

In order to send a DOS or OS/2 executable file received from a personal computer back to a personal computer, use the combined and no headers options:

```
BINSPEC BINARY(C) FORMAT(N)
```

Note: A variable length file sent with the above options cannot always be recreated, because the records have been combined and no record length information has been added. This format is needed to send data to some PC translators.

Sending Variable Length Files to PCs

In order to send a variable length file (such as a LIST3820) to a personal computer, so that none of the record length information is lost, even if all of the records are combined into one file, use the combined and any format option other than N.

```
BINSPEC BINARY(C) FORMAT(S)
```

In order to convert a file formatted as described above back to its original format, use the same options on receive processing.

Sending a File to a System with a Limit

Some systems have a limit to the size of a binary segment they can process. In order to limit the size of the data inside the BIN02 element to 1000 bytes, use the following option:

```
BINSPEC LIMIT(1000)
```

Note: A variable length file with records larger than 1000 bytes sent with the above options cannot always be recreated, because the records have been split and no record length information has been added. The format parameter could also be used to add record length information.

Mapping an Application Field to a Binary Segment

Although using a file name is the primary way of providing data for a binary segment, you may find it useful to provide the data by simply passing it to DataInterchange in application fields, the way all other application data is passed. If you do this, the application provides the length of the binary data and the binary data itself.

However, DataInterchange has a maximum length of 999 bytes for application fields, which is significantly less than the 999,999,999,999,999 defined by the standard or the 2,147,483,547 that DataInterchange allows. DataInterchange knows that it is dealing with binary data, and that some special processing is needed to determine the length of data being passed, gather it, and put it into a BIN segment. The easiest way to pass this binary data to DataInterchange is with a data format such as this:

BINARY_STRUCTURE1	ST	999	Repeating structure
LENGTH	N0	15	15-byte length field
BINARY_DATA1	ST	99	Repeating structure
BINARY_FIELD	CH	999	Binary data field

Both BINARY_STRUCTURE1 and BINARY_DATA1 are passed separately. Now the application provides DataInterchange with a BINARY_STRUCTURE1 and as many BINARY_DATA1 structures as necessary to satisfy the length specified in the LENGTH field. DataInterchange builds one BIN segment for each

Using the 841 Transaction Set

occurrence of BINARY_STRUCTURE1. The BIN segment has a length value as specified by the LENGTH field and contains data from the BINARY_DATA1 structures.

If the BINARY_DATA1 structures you provide exceed the length you specified, DataInterchange ignores the extras. If you provide fewer than needed, binary zeros are added to the BIN segment until the LENGTH value is satisfied. The number of repetitions you specify for the structures is not important. DataInterchange accepts as many as the application passes.

As another example, suppose the application wants to pass in data from the file and have structures built in the same way described above for variable-length files. The normal variable-length record format for MVS records is a binary halfword containing the length of the data followed by the data itself. The data format definition required to accomplish this is:

BINARY_STRUCTURE2	ST	999	Repeating structure
LENGTH	I0	2	2-byte binary length
BINARY_DATA2	ST	32756	Repeating structure
BINARY_FIELD	CH	1	Binary data field

BINARY_DATA2 is defined as physically part of its parent. Its maximum use count is 32756, because that is the maximum logical record length for MVS physical sequential files.

The only difference between this example and the previous one is in the way data is passed to DataInterchange. In the first case, BINARY_STRUCTURE1 is passed followed by multiple BINARY_DATA1 structures. In the second case, only BINARY_STRUCTURE2 is passed because BINARY_DATA2 is defined as physically part of BINARY_STRUCTURE2.

There is a trade-off in choosing between the two cases. Structures always have a fixed length in DataInterchange. Therefore, when DataInterchange is presented with BINARY_STRUCTURE2, it always expects to get 32758 bytes. On the receive side, it always creates 32758 bytes. In effect, each variable-length record is expanded to its maximum length. The method used with BINARY_STRUCTURE1 conserves main storage at the expense of making the application break up the data into 999-byte chunks, in order for DataInterchange to put it back together.

You may ask, "But a structure can repeat only 32757 times, and a field can be only 999 bytes long, for a maximum record size of 32,724,243 bytes? What if I want to pass in more than that as a single structure?" DataInterchange assumes that if the field mapped to the binary data element of the BIN segment is the first field of a structure, then the length of the *structure*, not the length of the field, is used to move data. Furthermore, if the structure is defined to be physically part of its parent, and its parent is defined to be physically part of its parent, and it has no other fields, DataInterchange uses the length of the parent structure. For example:

BINARY_STRUCTURE3	ST	999	Repeating structure
LENGTH	N0	15	15-byte length field
BINARY_DATA3	ST	10000	Repeating structure
BINARY_DATA3A	ST	10000	Repeating structure
BINARY_DATA3B	ST	10000	Repeating structure
BINARY_FIELD1	CH	999	Binary data field
BINARY_FIELD2	CH	1	Binary data field

BINARY_DATA3, BINARY_DATA3A, and BINARY_DATA3B are all defined to be physically part of their parent. Using this definition and mapping the binary data field in the standard to BINARY_FIELD1, the translator arrives at a length value of 1000 (the length of BINARY_FIELD1+BINARY_FIELD2) * 10000 (the number of times BINARY_DATA3B repeats) * 10000 (the number of times BINARY_DATA3A repeats) * 10000 (the number of times BINARY_DATA3 repeats). This yields an effective length of 10 to the 15th (1 greater than allowed by the standard).

Of course this structure could not really be used because the maximum length allowed by DataInterchange is 2,147,483,647. Nevertheless, the example illustrates the technique of creating an effective length greater than 999.

Receive Processing for the Binary Segment

This section covers mapping a binary segment to a file or an application field.

Mapping Data from a Binary Segment to a File

Receiving presents a set of problems that are not present on the send side. During send processing, it is reasonable to assume that senders know what they are sending, when they are sending it, and the data set names of the files they are sending. Receivers, on the other hand, may have little control over what they receive, when they receive it, and how many files they receive at any one time. Therefore, DataInterchange must be capable of creating files when necessary.

As with sending, if you want the binary data passed to you in a file rather than the application fields, map the binary data to a field in the application defined with the FN data type. You will not receive the binary data in the field; instead you will receive the name of the data set the data was written to.

On the send side, you can provide the file name in an application field buffer or as a literal value. You can also provide a literal on the receive side. If you provide a literal for an FN data type that is mapped to the binary data element of the BIN segment (BIN02), DataInterchange gives the literal special processing. Normally, a literal mapped for receiving is used only if the standard data element is blank. The special processing for this literal is that you can use it to provide the file name and allocation parameters that DataInterchange needs to create the file on your system.

The literal has this format:

type:name parameters

where:

type

is an optional value that indicates the type of file being provided. Valid values are:

Value	Description
DD	Data definition name (ddname)
DS	Data set name
MQ	MQSeries Queue profile member name
VS	VSAM entry sequenced data set
TD	Transient data queue
TM	Temporary storage queue (main)
TS	Temporary storage queue (auxiliary)

MQ, VS, TD, TM, and TS are for CICS only. The default is DS.

name

is either a ddname or data set name based on the value of *type*. It must conform to the conventions for its type. A ddname has a maximum length of 8 characters. A data set name has a maximum length of 44 characters.

name can also contain one of the following special literals to substitute other values in this field:

Using the 841 Transaction Set

Literal Substitutes

&IV	A value that represents the current binary file that is being processed. An <i>F</i> is inserted in front of this value if the IV value begins another level of qualification on the data set name.
&U	The current user ID. A period is inserted after the user ID to force another level of qualification on the data set name.
&D	The current date as a <i>yymmdd</i> value. A <i>D</i> is inserted in front of this value if it begins another level of qualification on the data set name.
&T	The current time as an <i>hhmmss</i> value. A <i>T</i> is inserted in front of this value if it begins another level of qualification on the data set name.
&E	The current file name from the EFI segment. Another level of qualification is forced both before and after the EFI file name.

name is optional. If you do not provide it, the file name from the EFI segment is used. If an EFI segment does not exist or does not contain a file name field, a default data set name &U.BIN&V.D&D.T&T is used. The name that is used is returned to the application in the FN field. The maximum length of a data set name is 44 characters, and truncation occurs if necessary.

parameters

is a series of keywords and values that supply the data needed to create a new data set on your system. Except for the first keyword (ALL), you can specify them in any order. The keywords are shown with an acceptable abbreviation in CAPS:

Keyword	Description																
ALLocate	Signals that allocation parameters follow.																
TRackS	Allocation unit is tracks.																
CYLinders	Allocation unit is cylinders. This is the default value if TRK or BLK is not specified.																
BLockS(<i>b</i>)	Allocation unit is BLOCKS with an average block length of <i>b</i> .																
Space(<i>p,s</i>)	Primary space quantity of <i>p</i> (default value of 10). Secondary quantity of <i>s</i> (default value of 10).																
UNit(<i>vvvv</i>)	<i>vvvv</i> is the unit name for the allocation. Default value is SYSDA.																
UCount(<i>n</i>)	<i>n</i> is the number of units for the allocation. Default value is 1.																
Lrecl(<i>l</i>)	<i>l</i> is the logical record length. Default value is 32756 or the value taken from the EFI segment.																
Blksize(<i>b</i>)	<i>b</i> is the block size. Default value is 32760 or the value taken from the EFI segment.																
Recfm(<i>xxxx</i>)	<i>xxxx</i> is the record format for records in the file. Default value is VB or the value taken from EFI segment. Valid values are: <table><tr><th>Value</th><th>Description</th></tr><tr><td>A</td><td>ASA printer control characters</td></tr><tr><td>B</td><td>Block</td></tr><tr><td>F</td><td>Fixed</td></tr><tr><td>M</td><td>Machine code printer control characters</td></tr><tr><td>S</td><td>Spanned</td></tr><tr><td>U</td><td>Undefined</td></tr><tr><td>V</td><td>Variable</td></tr></table>	Value	Description	A	ASA printer control characters	B	Block	F	Fixed	M	Machine code printer control characters	S	Spanned	U	Undefined	V	Variable
Value	Description																
A	ASA printer control characters																
B	Block																
F	Fixed																
M	Machine code printer control characters																
S	Spanned																
U	Undefined																
V	Variable																
LIKE(<i>dsname</i>)	<i>dsname</i> is the name of some other data set from which the LRECL, BLKSIZE, and RECFM are copied.																

DataInterchange first tries to open a given data set name. If the attempt to open the data set is successful, the file is used, and any data in the file is overlayed by the new data. If the attempt fails, DataInterchange tries to create a new file using the allocation value (or defaults) shown above. The new data set has a specification of (NEW,CATLG,CATLG), and all unused space in the data set is released when the data set is closed. For DI for CICS, if the attempt to open the data set fails, a unique temporary storage queue is created, and the data is written to this queue. Default values are overridden by values from the EFI segment, which in turn are overridden by literal values supplied at mapping time.

If the data being received is being written to a fixed-length file, DataInterchange assumes the binary segment consists of multiple records without any record headers. If the data being received is being written to a variable-length file, DataInterchange assumes each binary segment contains data for a single record without any record headers.

DataInterchange takes data from the binary segments and adds them to the file. For each BIN segment, the amount of data defined by the logical record length of the file is extracted and written to the file. This continues until the amount of data left in the BIN segment is less than or equal to the logical record length. If the amount of data left in a binary segment is less than the logical record length, fixed-length records are padded with binary zeros; variable-length records are written with the shortened logical record length. Data from the end of one BIN segment is never combined with the data from the beginning of the next BIN segment. At a minimum, each BIN segment defines the beginning of a new record in the file.

If the defaults previously described are not desired, then the binary specifications described in “Format Specifications” on page B-3 (BINary and Format) can be provided as part of the allocation parameters in the receive literal value.

Mapping Data from a Binary Segment to an Application Field

The considerations described for mapping from an application field to a binary segment also apply when mapping data from a binary segment to an application field. See “Mapping an Application Field to a Binary Segment” on page B-5.

Appendix C. The IBM DataInterchange Web Site

The IBM DataInterchange web site, located at <http://delilah.fl.us.ibm.net/datainterchange>, supplies support and information for DataInterchange users. In addition to information on year 2000 readiness and pageable translation, the site includes the following features:

1. **Fact Sheet - An overview of DataInterchange for MVS and DataInterchange for CICS.** This section describes:

- Interface performance
- Event-driven EDI functionality
- Client/server design
- Client/server functions
- Translation and transmission management features
- Domestic and international support
- Standards updates
- Online customization and maps
- Transaction management efficiency
- Comprehensive reporting and control
- Query function
- Application interface and CICS support
- Interactive entry facility
- Flexible communication options

2. **Details on the Annual DataInterchange User's Group Conference.** This three-day conference offers a comprehensive look at the latest functions and applications of DataInterchange, along with case studies and "how to" sessions.

3. **Technical Support Information.** This section lists technical support contact numbers by geographic area. A support survey allows you to evaluate the responses to your technical questions. A "Hot Stuff" option provides tips and techniques regarding common DataInterchange technical issues.

4. **FAQs.** Answers to frequently asked questions.

5. **Download Area.** The options for downloading include:

- DataInterchange Client and Host standards
- The latest DataInterchange Client FIXPAK
- Demonstration version of DataInterchange Client

Glossary and Index

Glossary

A

AAR. Association of American Railroads. Represents the railroad industry in areas such as standards, public relations, and advertising.

acknowledgment. See *functional acknowledgment*, *network acknowledgment*.

action bar. The area at the top of a panel that contains choices currently available in the application that is running. Compare to the *function key area*, which contains actions common to all programs.

ADF. See application data format.

ANSI. American National Standards Institute.

ANSI ASC X12. ANSI Accredited Standards Committee X12, which develops and maintains generic standards for business transactions for EDI.

application. A program that processes business information. An application that requests services from DataInterchange is an *enabled application*.

application data. The actual data in a transaction.

application data format. A description of the application data for a particular transaction. An application data format is composed of data structures and fields.

B

base structure. The data structure that contains all the data structures and data fields that define the application data for a single transaction.

binary format (BIN). Representation of a decimal value in which each field must be 2 or 4 bytes long. The sign (+ or –) is in the far left bit of the field, and the number value is in the remaining bits of the field. Positive numbers have a 0 in the sign bit. Negative numbers have a 1 in the sign bit and are in twos complement form.

C

CICS. Customer Information Control System.

CLIST. Command list.

command line. The line at the bottom of the panel that provides an alternate way of requesting services, rather than using the *Action* column of the panel body.

Command list (CLIST). A list of commands and statements designed to perform a specific function for the user.

composite data element. In EDIFACT standards, a group of related subelements, such as the elements that make up a name and address.

control string. A compiled map; a machine-readable representation of a map generation that optimizes performance during translation.

control structure. The beginning and ending segments (header and trailer) of standard enveloped transmissions.

Customer Information Control System (CICS). An IBM-licensed program that enables transactions entered at remote terminals to be processed concurrently by user-written application programs.

customize. To alter to suit the needs of a company, such as removing from an EDI standard the segments and data elements that the company does not use.

D

DASD. Direct access storage device.

data dictionary. A file containing the definitions of all the data elements of an EDI standard.

data element. A single item of data in a standard, such as a purchase order number. Corresponds to a data field in a data format.

data element delimiter. A character, such as an asterisk (*), that follows the segment identifier and separates each data element in a segment. See also *element separator* and *segment ID separator*.

data field. A single item of data in a data format, such as a purchase order number. Corresponds to a data element in a standard.

data format. See *application data format*.

data structure. A group of related data fields in a data format, such as the fields making up the line item of an invoice. Corresponds to a segment in a standard.

| **DB2.** Database 2, an IBM relational database management system.

ddname. Data definition name.

decimal notation. The character that represents a decimal point in an envelope standard.

| **DI Client.** DataInterchange Client, the Windows-based, client/server interface for DataInterchange.

direct access storage device (DASD). A device in which access time is effectively independent of the location of the data.

distribution libraries. Supplied partitioned data sets on tape containing one or more components used to transfer data to a new system.

distribution tape. A magnetic tape that contains the distribution libraries for installing a new system.

domain. The data structure or group of data structures in a data format to and from which you should restrict the mapping of EDI repeating segments and loops.

E

EDI. Electronic data interchange.

EDIA. Electronic Data Interchange Association.

EDI administrator. The person responsible for setting up and maintaining DataInterchange.

| **EDIFACT.** Electronic Data Interchange for Administration Commerce and Transport.

electronic data interchange (EDI). A method of transmitting business information over a network, between business associates who agree to follow approved national or industry standards in translating and exchanging information.

electronic transmission. The means by which information is transferred between parties, such as over a public network.

element. See *data element*.

element separator. A character that separates the data elements in a segment. See also *data element delimiter*.

encryption. The encoding and scrambling of data. Data is encrypted by the sender and decrypted by the

receiver using a predetermined program and unique electronic key.

event. An occurrence that is important to a user's computer task, such as a software error, sending a transaction, or acknowledging a message.

F

field. See *data field*.

functional acknowledgment. An electronic acknowledgment returned to the sender to indicate acceptance or rejection of EDI transactions.

functional group. One or more transaction sets of a similar type transmitted from the same location, enclosed by functional group header and trailer segments.

function key. A key that causes a specified sequence of operations to be performed when it is pressed. Generally used to refer to keys labelled F_n , where n is a number from 1 to 24.

function key area. Two lines at the bottom of the panel that list the active function keys for the panel.

G

GDDM. Graphical data display manager.

| **graphical data display manager (GDDM).** An IBM-licensed program that allows pictures to be defined and displayed on your monitor.

H

header. A control structure that indicates the start of an electronic transmission.

| **hierarchical loop.** A mapping construct in which a multi-generational parent/child structure of a repeating nature can be described; also called HL.

| **HL.** See hierarchical loop.

I

IBM Global Network. The worldwide IBM communications network that provides network solutions and a global information infrastructure through IBM and other companies.

ICS. International Control Segments

Interactive System Productivity Facility (ISPF). An IBM-licensed program that serves as a full-screen editor and dialog manager.

interchange. The exchange of information between trading partners.

ISPF. Interactive System Productivity Facility.

J

JCL. Job Control Language.

K

key. In a profile member, the field that identifies the member. For example, the key for members of the trading partner profile is the trading partner nickname.

L

Link Pack Area (LPA). In MVS, an area of main storage containing reenterable routines from system libraries. Their presence in main storage saves loading time.

literal. In transaction mapping, a value that is constant for each occurrence of the transaction. If you provide the literal value during mapping, the translator does not have to refer to an application field to obtain the value.

log file. A file in which events are recorded.

logging. The recording of events in time sequence.

loop. A group of related segments in a transaction set.

loop ID. A unique code identifying a loop and the number of times the group can be repeated.

loop repeat. A number indicating the maximum number of times a loop can be used in succession.

LPA. Link pack area.

M

| **map.** A set of instructions that describes how to transform application data into EDI data.

maximum use. A number indicating the maximum number of times a segment can be used in a transaction set.

member. A collection of data for one entry in a profile. For example, a member of the trading partner profile contains data about one trading partner.

message. A free-form, usually short, communication to a trading partner. In EDIFACT standards, a transaction set.

| **MQSeries.** A product of the IBM company; used to implement messaging and queueing of data groups.

multiple-occurrence mapping. A form of mapping in which all occurrences of a loop or repeating segment are mapped to the same repeating structure in the data format.

MVS. Multiple virtual storage.

N

network acknowledgment. A response from the network indicating the status of an interchange envelope, such as sent or received.

O

ODETTE. Organization for Data Exchange through Teletransmission in Europe.

P

panel body. The area in the middle of the panel that contains entry fields, lists of selectable items, menu choices, and scrollable text.

PDF. Program Development Facility.

PDS. Partitioned data set.

PDS members. Groups of related information stored in partitioned data sets.

| **profile.** A group of similar objects. Each profile can contain one or more objects or *profile members*. For example, the trading partner profile contains members for each of your trading partners (one member per specific trading partner). All members have predefined fields in which you may enter data. Each member of the trading partner profile, for example, has fields that will help you identify your trading partners (like company name and address). For some fields, such as the key field, the data that you enter *MUST* be different for each member. The first field of the profile member is the *key* or name for referring to the member. In the trading partner profile, for example, the key is the trading partner nickname.

program directory. A document shipped with each release of a product that describes the detailed content of the product.

Q

qualifier. A data element which gives a generic segment or data element a specific meaning. Qualifiers are used in mapping single or multiple occurrences.

quiesce. To end a process by allowing operations to complete normally.

quiescing. The process of bringing a device or a system to a halt by rejection of new requests for work.

R

RACF. Resource access control facility.

release character. The character that indicates that a separator or delimiter is to be used as text data instead of as a separator or delimiter. The release character must immediately precede the delimiter.

repository data. A group of data definitions, formats, and rules, that DataInterchange uses to process your data.

Resource Access Control Facility (RACF). An IBM-licensed program that provides for access control by identifying and verifying the users to the system, authorizing access to protected resources, logging the detected unauthorized attempts to enter the system, and logging the detected accesses to protected resources.

S

SAF. System Authorization Facility.

security administrator. The person who controls access to business data and program functions.

| **SAP.** A German company named Systeme,
| Anwendungen, and Produkte specializing in application
| software. A major product, SAP R/3, is a
| component-based architecture/application that
| integrates business processes such as sales, materials
| management, and distribution.

| SAP R/3 supports an EDI interface subsystem. SAP
| R/3 generates application data in the SAP R/3
| Intermediate Document (IDOC) layout. This data file is
| then sent to EDI subsystem via a file transfer product
| such as FTP or TCP/IP.

segment. A group of related data elements. A segment is a single line in a transaction set, beginning with a function identifier and ending with a segment terminator delimiter. The data elements in the segment are separated by data element delimiters.

segment directory. A file containing the format of all segments in a standard.

segment identifier. A unique identifier at the beginning of each segment consisting of two or three alphanumeric characters.

segment ID separator. The character that separates the segment identifier from the data elements in the segment.

segment terminator. The character that marks the end of a segment.

single-occurrence mapping. A form of mapping in which each occurrence of a loop or repeating segment is mapped to a different part of the data format.

SMP/E. System Modification Program Extended.

SQL. Structured query language.

standards. The industry-supplied, national, or international formats to which information is converted, allowing different computer systems and applications to interchange information.

structure. See *data structure*.

subelement. In EDIFACT standards, a data element that is part of a composite data element. For example, a data element and its qualifier are subelements of a composite data element.

subelement separator. A character that separates the subelements in a composite data element.

System Authorization Facility (SAF).

System Modification Program Extended (SMP/E). An IBM-licensed program used to install software and software changes on OS/VS1 and OS/VS2 systems.

T

tag. In EDIFACT standards, the segment identifier.

TDCC. Transportation Data Coordinating Committee.

TDQ. Transient data queue.

temporary storage queue (TS). Storage locations reserved for immediate results in CICS. They are deleted after the task that created them is complete and they are no longer necessary.

| **Time Sharing Option (TSO).** A component of the IBM
| MVS operating system which allows users full access to
| MVS functionality, but shares machine resources across
| users.

Time Sharing Option Extensions (TSO/E). The base for all TSO enhancements. It provides MVS users with additional functions, improved usability, and better performance.

| **TPT.** See trading partner transaction.

trading partner profile. The profile that defines your trading partners, including information about network account numbers, user IDs, who pays for network charges, etc.

trading partners. Business associates, such as a manufacturer and a supplier, who agree to exchange information using electronic data interchange.

trading partner transaction. A transaction set customized to match the format that two trading partners have agreed to use for exchanging one type of transaction.

trailer. A control structure that indicates the end of an electronic transmission.

transaction. A single business document, such as an invoice.

transaction set. A group of standard data segments, in a predefined sequence, needed to provide all of the data required to define a complete transaction, such as an invoice or purchase order.

transient data queue. A sequential data set used by the Folder Application Facility in MVS/CICS to log system messages.

translation. The process of converting information from a data format to a standard format, or from a standard format to a data format.

translation table. A user-defined table that translates data values that differ between trading partners. For example, if a manufacturer and supplier have different part numbers for the same item, each company can use its own part number and have it converted to the other company's part number during translation.

| **TSO.** See Time Sharing Option.

TSO/E. Time Sharing Option/Extended.

| **TSQ.** Temporary storage queue; a CICS mechanism for storing data within the CICS data space; also TS queue.

U

UCS. Uniform Communication Standard.

unary operator. An operator that changes the sign of a numeric value.

Uniform Communication Standard (UCS). The EDI standard used in the grocery industry.

UNTDI. United Nations Trade Data Interchange.

V

validation table. A table, supplied by DataInterchange or defined by the user, which contains all acceptable values for a single data field.

Virtual Storage Access Method (VSAM). An access method for direct or sequential processing of fixed and variable-length records on direct access devices. The records in a VSAM data set or file can be organized in logical sequence by a key field (key sequence), in the physical sequence in which they are written on the data set or file (entry-sequence), or by relative-record number.

VSAM. Virtual Storage Access Method.

W

WINS. Warehouse Information Network Standard.

X

| **X12.** ANSI ASC X12 standard used mostly in North American business.

Index

Special Characters

&ACFIELD 9-8
&ASSERT 9-8
&BAT_{nn} 9-34
&BG_{nn} 9-34
&D B-8
&DATE 9-8
&DEFERRED 9-9
&E 9-9, B-8
&EG_{nn} 9-34
&END_{nn} 9-34
&EOB_{nn} 9-34
&ERR 9-9
&FORCE 9-10
&FORMAT 9-10
&G 9-35
&GAR 9-35
&GAS 9-35
&GCN 9-35
&GDT 9-35
&GE_{nn} 9-34
&GFG 9-35
&GLV 9-35
&GPW 9-35
&GS_{nn} 9-33
>M 9-35
>T 9-35
&GVR 9-35
&HCHILD 9-109
&HCODE 9-109
&HLID 9-109
&HLPID 9-109
&I 9-34
&IAP 9-34
&ICE_{nn} 9-34
&ICN 9-34
&ICS_{nn} 9-34
&ICT 9-35
&IDT 9-34
&IEA_{nn} 9-34
&IF 9-10
&IFDATA 9-10
&IFNODATA 9-11
&IFNOVAR 9-11
&IGT 9-35
&IIR 9-34
&IIS 9-34
&IPW 9-34
&ISA_{nn} 9-33
&ITM 9-34

&ITT 9-35
&IV B-8
&IVR 9-35
&LIT 9-51, 9-52
&LOOPBREAK 9-11
&LOOPCHECK 9-11
&LSAVE 9-12
&LSET 9-13
&LSID 9-13
&MHD_{nn} 9-34
&MTR_{nn} 9-34
&SAMEAS 9-13
&SAVE 9-14
&SE_{nn} 9-33
&SET 9-14
&ST_{nn} 9-33
&STX_{nn} 9-34
&T 9-35, B-8
&TCN 9-35
&THANDLE 9-15
&TIME 9-15
&TLV 9-35
&TPID 9-15
&TPNICKN 9-15
&TTC 9-35
&TTS 9-35
&TVR 9-35
&U B-8
&UNB_{nn} 9-34
&UNE_{nn} 9-34
&UNG_{nn} 9-34
&UNH_{nn} 9-34
&UNT_{nn} 9-34
&UNZ_{nn} 9-34
&USE 9-15
&VAR 9-51, 9-52
&ZEROSIG 9-16
&, loop ids 6-18

Numerics

841 transaction set
 keywords B-8
 literals B-8

A

acceptable error level 9-84, 9-96
access to menu options, controlling A-2
accessing
 DataInterchange for CICS 2-3
 DataInterchange for MVS 2-1

account number 5-16, 5-28

accumulators

- actions 9-4
- description 9-3
- literals 9-6
- mapping 9-57
- names 9-4

Acknowledgment action 10-31

acknowledgment expected 9-81

acknowledgment request (ISA14) 6-57

acknowledgment type 9-91

acknowledgments, functional

- delayed enveloping for 5-50
- description 5-48
- receiving 5-49
- requesting 5-48
- sending 5-48

actions

- Acknowledgment 10-31
- Detail 10-31
- Envelope 10-32
- eVentlog 10-33
- eXclude 10-33
- grOups 10-33
- Hold 10-33
- Image 10-34
- Line 10-34
- List 2-10
- Log 12-9
- multiple
 - entering 2-7
 - order processed 2-8
- Options 10-34
- Print 2-11, 10-35
- purGe 10-36
- Release 10-36
- Report 10-36
- Send 10-36
- Summary 10-37
- Transactions 10-37
- Translate 10-37
- Unpurge 10-37

activating continuous receive 5-46

active usage

- receive transaction 9-93
- send transaction 9-81

activity log profile

- application ID 12-3
- description 12-1
- log file name 12-3
- log flag 12-3
- sample member of 12-9

ACTLOGS 12-1

ADAMCTL 4-5

adding

- application data formats 7-7
- business document definitions 13-4
- business documents 13-13
- file requests 13-35
- members
 - envelope profile 6-30
 - network profile 5-8
 - requestor profile 5-14
 - trading partner profile 5-23
- messages 13-28
- requests to send files 13-35
- segments 6-13
- trading partner usages
 - receive transaction 9-88
 - sending 9-78
- transaction sets 6-15
- translation tables 8-5
- validation tables 8-10

additional records 5-47

address line 2, in TPPROF 5-29

address line 3, in TPPROF 5-29

administrative data

- exporting 11-1
- importing 11-1

Administrator's Menu (MP01)

- controlling access to A-3
- illustration 2-5
- invoking the DataInterchange Utility 15-1
- resource names A-3

administrator, security A-1

agency code 9-90

agency code (AG) data type 6-33

agreements, trading partner 6-4

ALLocate keyword B-8

allocating data sets 3-6

allow generic recv 5-37

alphabetic (A) data type 6-21, 7-17

alphanumeric (AN) data type 6-21, 7-17

alphanumeric table (ALPHANUM) 8-3

alternate network dial number 5-22

ampersand, loop ids 6-18

ANSI

- where to write for information xxii
- X12 terms compared to EDIFACT xxiii

ANY keyword 5-3, 5-4

API (Application Programming Interface) 1-8

APPDEFS 4-11

application control (AC) data type 7-17

application control field 9-51

application criteria

- application control number 10-14
- application data format ID 10-14
- application ID 10-14
- batch ID 10-14
- date delivered to application 10-14

application criteria *(continued)*

time delivered to application 10-14

application data

defining the format of 7-1

logging 12-9

reformatting for C and D records 7-20

application data formats

adding 7-7

business document definitions 13-5

business documents 14-2

changing after mapping 7-23

create standard authority A-6

data types 7-17

defining 7-1

description 7-1, 7-8

Document Layout Utility 14-2

document layouts 14-2

example of 7-7

exporting 11-7, 11-9, 11-13, 11-14

importing 11-18

Prompt 2-14

resource names A-6

sample purchase order

receive transaction 9-47

send transaction 9-46

selecting 7-6

summary of tasks for 7-1

application definition profile (APPDEFS)

ACTLOGS member 4-12

application ID 4-12

description 4-11

functional acknowledgment image 4-13

log standard data 4-13

management reporting 4-12

monitor program 4-13

production usage - test message 4-13

test usage 4-13

transaction image 4-12

transaction store 4-12

application file

name in a data format 7-8

overriding name of 9-95

types for CICS 7-9, 9-95

application ID 4-12, 5-47, 12-3**application password**

overriding

receive transactions 9-94

send transactions 9-83

UNG13 6-40

Application Programming Interface 1-8, 15-1**application receiver**

BG04 6-51

GS03 6-44, 6-52, 6-58

overriding

receive transactions 9-90

send transactions 9-83

application receiver (AR) data type 6-33**application reference (AP) data type 6-33****application reference (STX11) 6-49****application sender**

BG03 6-51

GS02 6-44, 6-52, 6-58

overriding

receive transactions 9-90

send transactions 9-83

application sender (AS) data type 6-33**archiving**

business documents 13-19

event log records A-8

log entries

authority for A-8

arithmetic operators 9-18**ASSERTLVL keyword 9-8****associated objects for export**

business document definitions

application data formats 11-13

business document layouts 11-13

requestor profile 11-13

control strings

envelope profile 11-12

trading partner profile 11-12

trading partner usages 11-11

translation exit routines 11-12

translation tables 11-11

user exit routines 11-12

validation tables 11-11

document layouts for application data

formats 11-14

standards

envelope profile 11-6

envelope standard 11-6

transaction sets 11-6

validation tables 11-6

trading partner transactions

application data formats 11-9

control strings 11-9

envelope profile 11-9

mappings 11-9

standard transactions 11-9

trading partner profile 11-9

trading partner usages 11-9

translation exit routines 11-9

translation tables 11-9

user exit routines 11-9

validation tables 11-9

association assigned

UNG12 6-39

UNH06 6-41

authentication

code 4-10

exit routine 4-5

program 4-10

authentication (*continued*)

type 4-10

authorization

information (ISA02) 6-55

information qualifier (ISA01) 6-55

verification 5-19

automating receiving 5-43

B

backward scrolling 2-10

base structure name 7-8

beginning data structure name 7-9

BIN segment

ID B-1

length B-1

receive processing B-7

send processing B-2

binary (Bn) data type 7-17

Blksize keyword B-8

BLocks keyword B-8

Boolean operators 9-18

buffer size 4-11

business document definitions

adding 13-4

application data format 13-5

exporting 11-12

files

archived documents 13-5

formatted documents 13-5

unformatted documents 13-5

importing 11-21

updating 13-6

viewing 13-6

business document layouts, exporting 11-13

business documents

adding 13-13

archiving 13-19

copying 13-18

customizing 14-1

deleting 13-18

description 13-9

filing 13-19

moving through panels 13-13

panel titles 13-10

panels, moving through 13-13

printing 13-22

recalling from network 13-24

receiving 13-25

restoring 13-19

sending 13-17

status codes for 13-10

translating 13-26

updating 13-22

viewing 13-23

C

C records

description 7-2

reformatting data for 7-20

Cancel key 2-11

changing

data formats after mapping 7-23

EDI standards 6-4

envelope standards 6-26

CHAR operator 9-19

character (CH) data type 7-17

character set table (CHARSET) 8-3

characters, single or double byte 4-3

city, in TPPROF 5-29

class of information 5-17

client/server mode

creating profiles 1-16

description 1-14

moving standards, data formats, or maps 1-16

updating profiles 1-16

code page ID 4-3, 4-4

Comm-Press** 5-37

command line on panels 2-7

command prompt 2-7

commands, entering 2-8

common access reference

MHD04 6-49

UNH07 6-41

common data header 5-20

communication routine 4-5, 5-10

communications ID (BG01) 6-50

communications password (BG02) 6-50

communications resource names A-7

communications with trading partners 5-1

company address, in TPPROF 5-29

company name (CO) data type 6-33

company name, in TPPROF 5-29

comparison operators 9-18

components, Interactive Entry Facility 13-2

composite element (CD) data type 6-21

composite elements

adding 6-9

updating 6-11

compression 5-22, 5-37

Comm-Press** software program 5-37

exit routine 4-5

program 4-11

sending data to network 5-37

concatenating data in data elements 9-61

conditional processing literals 9-7

connecting to networks

GEIS 5-4

IBM Global Network 5-5

others 5-8

- contact fax 5-30**
 - contact fax, in TPPROF 5-30
- contact name, in TPPROF 5-30**
- contact phone, in TPPROF 5-30**
- Continuous Receive Facility 5-43**
- continuous receive ID 5-46**
- continuous receive profile**
 - active 5-46
 - adding a member 5-44
 - additional records 5-47
 - allow SYNCPOINTS 5-47
 - application ID 5-47
 - continuous receive ID 5-46
 - develope only 5-47
 - delay functional acknowledgments enveloping 5-47
 - description 5-44
 - duplicate envelopes 5-48
 - exception name 5-46
 - exception type 5-47
 - FA TS queue 5-47
 - message user class 5-46
 - national language code 5-47
 - network acknowledgments 5-48
 - pageable 5-48
 - print name 5-46
 - print type 5-46
 - purge interval 5-48
 - raw data 5-46
 - requestor ID 5-46
 - response name 5-47
 - response type 5-47
 - track SAP status 5-48
 - trading partner nickname 5-46
 - translate 5-46
 - user field 5-47
- CONTRL message 5-48**
- control (C) records**
 - description 7-2
 - reformatting data for 7-20
- control data literals 9-33**
- control fields in document layouts 14-9**
- control number (CN) data type 6-33**
- control numbers**
 - group 5-40
 - special codes 5-40
 - transaction 5-40
- control standard identifier (ICS02) 6-42**
- control strings**
 - exporting 11-9, 11-10
 - generating 9-103
 - importing 11-20
- control version number (ICS03) 6-42**
- controlling access to menu options A-2**
- controlling agency**
 - UNG09 6-39
 - UNH05 6-41

- converting date formats to standard 9-60**
- copying**
 - business documents 13-18
 - EDI standards 6-7
 - messages 13-30
- count (CT) data type 6-34**
- country, in TPPROF 5-30**
- create standard authority A-6**
- creating**
 - maps 9-45
 - trading partner transactions 9-45
- cursor position, initial 2-9**
- customized documents**
 - description 14-1
 - predefined 14-16
 - printing in batch 14-7
 - printing page headings on 14-13
- customizing**
 - business documents 14-1
 - document layouts 14-1
 - EDI standards 6-4
 - envelope standards 6-26
 - language 4-2
- CYLinders keyword B-8**

D

- D records**
 - description 7-2
 - reformatting data for 7-20
 - suggested uses 8-1
- data (D) records**
 - description 7-2
 - reformatting data for 7-20
- data elements**
 - adding 6-9
 - characteristics 6-2
 - concatenating data 9-61
 - data types
 - EDI standards 6-20
 - envelope standards 6-32
 - delimiter 5-35
 - description 6-1
 - mapping 9-3
 - mapping application fields 9-57
 - mapping to multiple fields 9-41
 - mapping with special handling 9-59
 - multiple occurrences of 9-40
 - qualifying 9-71
 - viewing 6-23
- data extracts**
 - Management Reporting 1-11
 - network activity 1-11
 - trading partner capability 1-11
 - trading partner profile 1-11
 - transaction activity 1-12

data flow through Transaction Store 10-2

data format ID 7-8

data formats

- adding 7-7
- business documents 14-2
- changing after mapping 7-23
- data types 7-17
- defining 7-1
- description 7-1, 7-8
- Document Layout Utility 14-2
- document layouts 14-2
- example of 7-7
- selecting 7-6
- summary of tasks for 7-1

data line timeout 5-22, 5-36

data overlay check 9-96

data structure names

- adding a structure 7-12
- beginning, for raw data 7-9
- ending, for raw data 7-10

data structures

- data type 7-14
- hierarchy 7-3
- hierarchy of 9-38
- passed separately 7-14, 9-38

data types

- agency code (AG) 6-33
- alphabetic (A) 6-21, 7-17
- alphanumeric (AN) 6-21, 7-17
- application control (AC) 7-17
- application receiver (AR) 6-33
- application reference (AP) 6-33
- application sender (AS) 6-33
- binary (Bn) 7-17
- character (CH) 7-17
- company name (CO) 6-33
- composite element (CD) 6-21
- control number (CN) 6-33
- count (CT) 6-34
- date (DT) 6-21, 7-18
- decimal (Ln) 7-18
- file name (FN) 7-18, B-2
- fixed standard 7-17
- for application data formats 7-17
- for EDI standards 6-20
- for envelope standards 6-32
- functional group (FG) 6-34
- hexadecimal (Hn) 7-18
- hexadecimal (HX) 7-18
- identifier (ID) 6-21, 7-18
- incrementing value (IV) 6-21, 6-34
- integer (In) 7-18
- integer (IT) 7-18
- interchange receiver (IR) 6-34
- interchange sender (IS) 6-34
- numeric (N) 6-21, 7-18

data types (continued)

- numeric (Nn) 6-21, 7-19
- packed decimal (PD) 7-19
- packed decimal (Pn) 7-19
- password (PW) 6-34
- real (R) 6-21, 7-19
- real (Rn) 6-21, 7-19
- release level (LV) 6-34
- routing receiver (RR) 6-34
- routing sender (RS) 6-34
- structure (ST) 7-19
- test indicator (TI) 6-34
- time (TM) 6-21, 7-19
- total segments (TS) 6-34
- total transactions (TT) 6-35
- transaction code (TC) 6-34
- version (VR) 6-35
- zoned decimal (ZD) 7-20
- zoned decimal (Zn) 7-20

data, entering 2-9

DataInterchange Client

- client/server mode 1-14
- databases 1-15
- Graphical User Interface, usage with 1-13
- interface
 - description 1-13
 - functions unavailable 1-15
- moving
 - data formats 1-16
 - maps 1-16
 - setup profiles 1-16
 - standards 1-16
 - trading partner profiles 1-16
- naming conventions 1-15
- server mode 1-14

DataInterchange for CICS, logging on 2-3

DataInterchange for MVS

- accessing 2-1, 2-3
- logging on 2-3
- logon options
 - export/import file 2-2
 - language ID 2-2
 - language profile 2-2
 - print 2-1
 - report file 2-1
 - system ID 2-1
- utility 15-1

DataInterchange Version 2 Release 1

enhancements xxviii

DataInterchange Version 3 Release 1

enhancements xxvii

DD01–Business Document Definitions 13-4, 13-7

DD02–Add Business Document Definition 13-5

DD03–Delete Business Document Definition 13-7

- DD04–Update Business Document Definition** 13-8
- DD05–View Business Document Definition** 13-8
- DE01–Business Documents** 13-10
- DE02–Select Data Format** 13-13
- DE03–Add business document** 13-15
- DE05–Delete business document** 13-18
- DE06–Update business document** 13-22
- DE07–View business document** 13-24
- DE08–Archive business document** 13-20
- DE09–File business document** 13-19
- DE10–Status Choices** 13-16, 13-23
- DE11–Network Acknowledgments Send** 13-17
- DE12–Receive Business Document** 13-25
- DE13–File formatted business document** 13-21
- DE14–Network Access** 13-17
- DE15–File Options** 13-19
- DE16–Network Acknowledgments Recall** 13-24
- DE17–Internal Trading Partner ID Selection** 13-14
- DE18–View Customized Document** 13-24
- DE19–Enter Document Key** 13-14
- DE20–Specify File to Restore** 13-21
- DE20–Trading Partner Selection** 13-14
- DE21–Archived Documents** 13-22
- DE22–Select Business Document Action** 13-9
- decimal (Ln) data type** 7-18
- decimal notation** 4-5, 5-35
- deenvelope only, in CONTRECV** 5-47
- default selection criteria** 10-8
- defining**
 - application data formats 7-7
 - event logs 12-9
 - maps 9-47
 - trading partner transactions 9-47
- delay functional acknowledgments enveloping** 5-47
- delayed functional acknowledgments** 5-50
- deleting**
 - business documents 13-18
 - file requests 13-37
 - messages 13-31
- delimiters, overriding** 5-35
- destination verification** 5-19, 5-32
- Detail action** 10-31
- determining trading partners** 5-42
- DF01–File Requests** 13-33
- DF02–Add Request to Send a File** 13-35
- DF03–Delete File Request** 13-37
- DF04–View File Request** 13-37
- DF05–Status Choices** 13-35
- DF06–Network Acknowledgments Send** 13-36
- DF07–Receive File** 13-38
- DF08–Network Access** 13-36
- DF09–Network Acknowledgments Recall** 13-38
- DF12–Select File Requests Action** 13-33
- DI Variables**
 - DIAPPFIL 9-24
 - DIAPPTYPE 9-24

- DI Variables** *(continued)*
 - DIAUTOCC 9-24
 - DICCCTRL 9-24
 - DICUSERDATA 9-24
 - DIERRFILTER 9-25
 - DIEXPTRACE 9-25
 - DIMAPCHAIN 9-25
 - DIMAPSWITCH 9-25
 - DISAPSEQ 9-25
 - DIVALLEVEL 9-25
 - DIVATYPE 9-26
 - DIVARTRACE 9-26
- dial connect num** 5-13
- DIAPPFIL 9-24**
- DIAPPTYPE 9-24**
- DIAUTOCC 9-24**
- DICCCTRL 9-24**
- DICUSERDATA 9-24**
- DIERRFILTER 9-25**
- DIEXPTRACE 9-25**
- DIMAPCHAIN 9-25**
- DIMAPSWITCH 9-25**
- DISAPSEQ 9-25**
- DIVALLEVEL 9-25**
- DIVATYPE 9-26**
- DIVARTRACE 9-26**
- DM01–Messages** 13-27
- DM02–Add Message** 13-28
- DM04–Delete Message** 13-31
- DM06–View Message** 13-31
- DM07–Status Choices** 13-29
- DM08–Network Acknowledgments Send** 13-17, 13-29
- DM10–Receive Message** 13-32
- DM11–Network Access** 13-30
- DM12–Network Acknowledgments Recall** 13-32
- DM15–Select Messages Action** 13-26
- DOCMAP 14-7**
- Document Layout Utility**
 - application data format 14-2
 - inputs 14-2
 - layout statements 14-1
 - output 14-7
 - starting 14-1
- document layouts**
 - control fields 14-6
 - customized 14-1
 - description 14-1
 - example
 - application data format 14-7
 - sample purchase order 14-8
 - exporting 11-14
 - fields more than one line 14-10
 - IEF control fields in 14-9
 - importing 11-22
 - messages 14-20

document layouts *(continued)*

- predefined
 - invoice 14-18
 - purchase order 14-16
- print file 14-7
- repeating structures 14-12
- statements 14-2
- suppressing
 - fields 14-9
 - groups of fields 14-11
 - literals 14-11

double-byte characters 4-3

duplicate envelopes 5-48

E

EDI

- definition 1-1
- Gateway, connecting to 5-8
- option 5-19, 5-32
- processing override 5-19, 5-32
- standards
 - copying 6-7
 - customizing 6-1, 6-4
 - data types 6-20
 - importing 3-1
 - requesting electronically 3-1
 - resource names A-6

EDI.SIDPRT 14-7

EDIA standards organization address xxii

EDIBTCH 5-4

EDIFACT

- CONTRL message 5-48
- terms compared to ANSI X12 xxiii
- where to write for information xxii

EDILTDB2 14-1

EDILTVSM 14-1

EDIQDAT file 5-11

EDIQDATE file 5-11

EDIQDATU file 5-11

EDIUDB2 15-1

EDIUVSM 15-1

EDIW 15-1

EFI segment B-1

EI00—Export Menu

- illustration 11-4
- options 11-4

EI01—Import Menu

- illustration 11-16
- options 11-17

EI02—Export Standards 11-5

EI03—Associated Objects for Standards 11-5

EI04—Standard Transaction Sets 11-6

EI05—Export Application Data Formats 11-7

EI06—Export TP Transactions 11-8

EI07—Associated Objects for TP Transactions 11-8

EI08—Trading Partner Usage for Sending 11-10

EI09—Trading Partner Usage for Receiving 11-10

EI10—Export Control Strings 11-11

EI11—Associated Objects for Control Strings 11-11

EI12—Export Business Document Definitions 11-13

EI13—Associated Objects for Document

Definitions 11-13

EI14—Export Business Document Layouts 11-14

EI15—Associated Objects for Document

Layouts 11-14

EI16—Import Standards 11-17

EI17—Import Application Data Formats 11-19

EI18—Import TP Transaction 11-19

EI19—Import Control Strings 11-21

EI20—Import Business Document Definitions 11-21

EI21—Import Business Document Layouts 11-22

EI22—Duplicate Object 11-17

EI23—Export Profiles 11-15

EI24—Profile Members 11-15, 11-23

EI25—Import Profiles 11-22

EI27—Export Tables 11-16

EI28—Import Tables 11-24

EI30—TP Usage Association 11-20

EL01—Event Logging 12-4

EL02—Selection Criteria 12-4

EL03—View Log Entry 10-33, 12-6

EL04—Log Data 12-6

electronic data interchange 1-1

element ID 6-12

end of text/message 5-34

ENDGROUP statement 14-4

ending data structure name 7-10

enhancements for Version 2 Release 1 xxviii

entering

- commands 2-8
- data 2-9
- multiple actions 2-8

entry fields 2-9

Envelope action 10-32

envelope criteria

- application receiver 10-13
- application sender 10-13
- date enveloped 10-13
- envelope type 10-13
- group control number 10-13
- interchange control number 10-13
- interchange receiver 10-13
- interchange sender 10-13
- time enveloped 10-13
- transaction control number 10-13

envelope exit routine 4-6

envelope profile

- description 6-30
- exporting 11-6, 11-9, 11-12
- generic 6-30

- envelope profile** *(continued)*
 - suffix 5-37
- envelope profile member, overriding** 9-83, 9-94
- envelope standards**
 - customizing 6-26
 - data types 6-32
 - definition of 3-1
 - envelope switch 9-92
 - exporting 11-6
 - resource names A-7
- envelope type, overriding** 1-7, 9-82
- envelopes** 6-3
- enveloping and sending transactions** 10-16
- enveloping of functional acknowledgments, delaying** 5-47
- enveloping transactions** 10-15
- errors acceptable during translation** 9-84, 9-96
- event logs**
 - authority for A-5
 - defining 12-9
 - description 12-1
 - printing 12-4
 - turning on and off 12-8
 - viewing 12-4
- eVentlog action** 10-33
- exchanging setup data** 11-1
- eXclude action** 10-33
- Exit key** 2-11
- exit routines**
 - post-translation 9-80
 - pre-translation 9-91
- Expedite Base/CICS, connecting to** 5-5
- Expedite Base/MVS, connecting to** 5-5
- export report** 11-24
- exporting**
 - administrative data 11-1
 - authority for A-8
 - business document definitions 11-12
 - control strings 11-10
 - document layouts 11-14
 - facility vs. utility 11-25
 - limitations 11-3
 - maps 11-7
 - overview 11-1
 - profiles 11-14
 - report 11-24
 - resource names A-8
 - restrictions 11-3
 - standards 11-5
 - tables 11-16
 - trading partner transactions 11-7
 - translation exit routines 11-9, 11-12
 - user exit routines 11-9, 11-12
- expression syntax** 9-17

F

- field exit routine** 4-5
- FIELD statement** 14-5
- field/structure name** 7-14
- fields**
 - in document layouts
 - longer than one line 14-10
 - suppressing 14-9
 - suppressing groups of 14-11
 - length in data formats 7-14
 - type 7-14
- file name (FN) data type** 7-18, B-2
- file name table (FILENAME)** 8-3
- file requests**
 - adding 13-35
 - deleting 13-37
 - description 13-33
 - recalling from network 13-38
 - receiving 13-38
 - sending 13-36
 - status codes for 13-34
 - updating status of 13-39
 - viewing 13-37
- file suffix** 5-37
- files**
 - application
 - name in a data format 7-8
 - types for CICS 9-95
 - business documents
 - archived documents 13-5
 - formatted documents 13-5
 - unformatted documents 13-5
 - EDIQDAT 5-11
 - EDIQDATE 5-11
 - EDIQDATU 5-11
 - for export/import 11-3
 - layout statements 14-14
 - QDATA 5-11
 - QDATAE 5-11
 - QDATAU 5-11
 - types for CICS 7-9
- filing business documents** 13-19
- filtering**
 - exit routine 4-5
 - program 4-11
 - routines 4-11
 - type 4-10
- first and last message**
 - MHD06 6-50
 - UNH09 6-41
- Fixed translation**
 - application format (element) 6-10
 - application format (segment) 6-14
 - application format (standard) 6-23
 - create standard 7-8

Fixed translation *(continued)*

- data types 6-10, 7-17
- fixed standard defined 6-23
- where data written 7-8

Fixed-to-Fixed (translating application data to application data) 1-9, 7-23

floating segments, mapping 9-41

flowcharts, DataInterchange processing 9-118

fold character 4-5

format of application data 7-1

format override 5-20, 5-33

formatting application data

- C and D records 7-20

forward scrolling 2-10

function keys 2-7, 2-9—2-14

functional acknowledgments

- delayed enveloping for 5-50
- description 5-48
- pending 10-11
- receiving 5-49
- requesting 5-48
- sending 5-48
- temporary storage queue 5-47

functional group

- data type (FG) 6-34
- date (UNG06) 6-39
- ID
 - GS01 6-44, 6-52, 6-58
 - in standard transaction 6-17
- time (UNG07) 6-39

functional groups

- control number
 - GE02 6-45, 6-53, 6-59
 - GS06 6-45, 6-53, 6-59
- creating for trading partner profile 5-35
- header and trailer 6-3
- reference number
 - UNE02 6-40
 - UNG08 6-39
- total in interchange
 - EG02 6-52
 - ICE01 6-43
 - IEA01 6-57

G

GEIS network, connecting to 5-4

generating control strings 9-103

generic envelope profiles 6-30

generic mapping

- for receive mapping 9-113
- for send mapping 9-113

generic receive usages

- considerations
 - export/import 9-97
 - management reporting 9-97
 - transaction store 9-97

generic receive usages *(continued)*

- definition of 9-96
- determining the trading partner 9-96
- functional acknowledgment processing 9-97
- selecting 9-97

generic routing 7-11

generic send usages

- considerations
 - export/import 9-88
 - management reporting 9-88
 - transaction store 9-88
- definition of 9-85
- determining the trading partner 9-86
- selecting
 - send usage 9-86
 - standard envelope profile member 9-87

getting

- help 2-9
- standards electronically 3-1

group

- authentication key name 9-81
- control number, in adding trading partner profile member 5-36
- control numbers 5-40—5-41
- encryption key name 9-80
- level FA only 9-92
- security profile member name, overriding 9-83

GROUP statement 14-3

grOups action 10-33

H

Harbinger InTouch* Gateway 5-4

help, requesting 2-9

hexadecimal (Hn) data type 7-18

hexadecimal (HX) data type 7-18

hierarchical loops

- description 9-105
- mapping 9-110
- mapping example 9-116, 9-117
- preparing to map 9-107

hierarchy of data structures 7-3, 9-38

HL segment

- description 9-106
- literal keywords 9-109
- mapping 9-110

Hold action 10-33

I

I envelope profile 6-42

IBM Global Network, connecting to 5-5

ICSKEY 6-42

identifier (ID) data type 6-21, 7-18

IEF

- components 13-2

IEF *(continued)*

- description 13-1
- prerequisites 13-3
- resource names A-4

IINB1, IINB41, IINB42, IINCICS, IINR3

connections 5-5

import report 11-24

importing

- administrative data 11-1
- application data formats 11-18
- authority for A-8
- business document definitions 11-21
- control strings 11-20
- document layouts 11-22
- facility vs. utility 11-25
- limitations 11-3
- maps 11-19
- overview 11-1
- profiles 11-22
- report 11-24
- resource names A-8
- restrictions 11-3
- standards 3-1, 11-17
- tables 11-23
- trading partner transactions 11-19

IN special operator 9-20

inbound envelope used on FA 9-92

incrementing value (IV) data type 6-21, 6-34

information

- classification 5-17
- receiver ID (ICS07) 6-43
- sender ID (ICS05) 6-42

input record length 5-10

integer (In) data type 7-18

integer (IT) data type 7-18

Interactive Entry Facility (IEF)

- components 13-2
- description 13-1
- prerequisites 13-3
- resource names A-4

interchange

- acknowledgments 5-50
- control 5-36
- control header and trailer 6-4
- control number
 - ICE02 6-44
 - ICS10 6-43
 - IEA02 6-58
 - ISA13 6-57
- control reference
 - STX09 6-48
- date
 - BG05 6-51
 - GS04 6-44, 6-53, 6-58
 - ISA09 6-56
 - STX07 6-48

interchange *(continued)*

- ID 5-29
- ID qualifier
 - ISA05 6-55
 - ISA07 6-55
- qualifier 5-29
- receive password 5-30
- receiver
 - data type (IR) 6-34
 - ID (ISA08) 6-56
- send password 5-30
- sender
 - data type (IS) 6-34
 - ID (ISA06) 6-55
- standards (ISA11) 6-56
- time
 - BG06 6-51
 - GS05 6-44, 6-53, 6-58
 - ISA10 6-56
 - STX08 6-48
- version (ISA12) 6-56

internal trading partner ID

- receive usage 9-91
- send usage 9-80

InTouch* Gateway

- communication routine
 - TIGCMD31 5-4
 - TIGCMV31 5-4
- network profile members
 - TIGD31 5-4
 - TIGV31 5-4
- network program
 - EDIBTC 5-4

invoice document layout 14-18

invoking the DataInterchange utility

- from Administrator's menu 15-2

IS special operator 9-21

K

key name

- group authentication 9-81
- group encryption 9-80
- transaction authentication 9-81
- transaction encryption 9-81

Keys help 2-7

keys, function 2-7

keywords

- 841 transaction set B-8
- examples 9-27—9-33
- HL segment 9-109
- mapping envelope data 9-34
- mapping techniques 9-26
- service segments 9-33—9-36

KNOWN keyword 5-3, 5-4

L

LANGPROF 4-3

language

language profile (LANGPROF)

- code page ID 4-4
- date mask 4-4
- decimal notation 4-5
- description 4-3
- fold character 4-5
- language profile ID 4-4
- negative sign 4-5
- time mask 4-4

language support

- alphanumeric table (ALPHANUM) 8-3
- character set table (CHARSET) 8-3
- customizing 4-2
- file name table (FILENAME) 8-3
- monocase table (MONOCASE) 8-3
- program name table (PRGMNAME) 8-3
- translation tables, customizing 4-3
- validation tables, customizing 4-3

LAYOUT statement 14-3

layout statements 14-1

- ENDGROUP 14-4
- FIELD 14-5
- GROUP 14-3
- LAYOUT 14-3
- LITERAL 14-5
- sample file 14-14
- STRUCT 14-5
- syntax 14-2

length

- BIN segment B-1
- fields in data formats 7-14

length of records 5-12

LIKE keyword B-8

limitations, export/import 11-3

Line action 10-34

List action 2-10

literal keywords

- examples 9-27—9-33
- HL segment 9-109
- mapping envelope data 9-34
- mapping techniques 9-26
- service segments 9-33—9-36

LITERAL statement 14-5

literals

- accumulators
 - &Gn 9-6
 - &Tn 9-6
- conditional processing
 - &ACFIELD 9-8
 - &ACFIELDn 9-8
 - &DATE 9-8
 - &DEFERRED 9-9
 - &E 9-9

literals (continued)

conditional processing (continued)

- &ERR 9-9
- &FORCE 9-10
- &FORMAT 9-10
- &IF 9-10
- &IFDATA 9-10
- &IFNODATA 9-11
- &IFNOVAR 9-11
- &LSAVE 9-12
- &LSET 9-13
- &LSID 9-13
- &SAMEAS 9-13
- &SAVE 9-14
- &SET 9-14
- &THANDLE 9-15
- &TIME 9-15
- &TPID 9-15
- &TPNICKN 9-15
- &USE 9-15
- &ZEROSIG 9-16
 - description 9-7
- control data 9-33
- description 9-4
- examples 9-27—9-33
- expressions 9-17
- for receive mapping 9-6
- for send mapping 9-5
- format of data in 9-6
- insert function 9-29
- keywords 9-8—9-16
- mapping
 - envelope data 9-33
 - service segments, generic 9-34
 - service segments, specific 9-33
- named variables 9-16
- operators 9-20
- segment creation 9-5
- suppressing, in document layouts 14-11

load module name 4-6

Log action 12-9

log application data 9-81, 12-9

log entries

- interpreting 12-7
- printing 12-4
- viewing 12-4

log file information profile (LOGDATA)

log file name 12-3

log flag 12-3, 12-8

log standard data 5-35

log standard data field 12-9

logging

- accesses to profiles 12-9
- application data 12-9
- events 12-1
- standard data 12-9

logging *(continued)*

- turning on and off 12-8

logon on

- DataInterchange for CICS 2-3

- DataInterchange for MVS 2-3

- logon options 2-1

logon actions

- for DataInterchange for CICS 2-3

- for DataInterchange for MVS 2-1

loop

- ID 6-17

- ids

- type 1 6-18

- type 2 6-18, 6-19—6-20

- qualifier 9-71

- repeat 6-17

loops

- hierarchical 9-105

- mapping 9-39

- qualifying 9-69

Lrecl keyword B-8**M****machine type 5-33****mailbox**

- See requestor profile

Management Reporting

- activating 4-12

- data extracts 1-11

managing transactions 10-22**mapping**

- accumulators 9-57

- authority for migrating A-8

- data elements

- application fields 9-57

- concatenating data 9-61

- special handling 9-59

- definition of 9-1

- exporting 11-9

- floating segments 9-41

- guidelines for 9-38

- hierarchical loops 9-110

- HL segment 9-110

- literals 9-58

- loops 9-39

- migrating a transaction A-8

- multiple fields to one data element 9-41

- named variables 9-65

- nonrepeating segments 9-39

- one data element to multiple fields 9-41

- one field to multiple data elements 9-41

- overview 9-2

- preparing the worksheet 9-43

- prerequisites 9-2

- Prompt 2-14

mapping *(continued)*

- repeating segments 9-39

- segments 9-52

- specifying occurrences 9-67

- specifying trading partners 9-78

- substringing data 9-64

- techniques for literal keywords 9-26

- validation during 9-37

- worksheet for sample purchase order 9-44

maps

- creating 9-45

- defining 9-47

- description 9-1

- exporting 11-7

- importing 11-19

maximum use

- data structures 7-13

- segments 6-17

member, profile 4-1**menus**

- Administrator's Menu (MP01) 2-5

- controlling access to options A-2

- Expedite/CICS (EXPMMS) 3-8

- Information Exchange Administration Services

- (M0000US) 3-3

- Interactive Entry Facility (MM01) 13-3

- Transaction Store Facility (TF01) 10-6

message

- handler 5-13

- line on panels 2-7

- processing program 4-5

- reference

- MHD01 6-49

- UNH01 6-40

- UNT02 6-41

- release

- UNG11 6-39

- UNH04 6-40

- text header 5-12

- type

- MHD02 6-49

- UNH02 6-40

- user class

- adding a continuous receive profile 5-46

- adding a requestor profile 5-17

- receiving transactions 10-19

- version

- MHD03 6-49

- UNG10 6-39

- UNH03 6-40

messages

- adding 13-28

- copying 13-30

- deleting 13-31

- description 13-26

- error, from Document Layout Utility 14-20

messages *(continued)*

- printing 13-31
- recalling from network 13-31
- receiving 13-32
- sending 13-29
- status codes for 13-27
- total in functional group (UNE01) 6-40
- total in interchange (END01) 6-49
- updating 13-30
- updating status of 13-32
- viewing 13-31

migrating mappings, authority for A-8

minimal trading partners 5-2

MM01—Interactive Entry Facility menu

- business document definitions option 13-4
- business documents option 13-4
- file requests option 13-4
- illustration 13-3
- messages option 13-4

mnemonics for actions 2-7

monitoring exit routine 4-5

monocase table (MONOCASE) 8-3

moving from test to production 11-1

MQSeries queue profile 4-7, 5-1, A-5

MQSeries queues 5-1, 5-13, 10-18, 10-21, 15-2

multiple

- environments with trading partners 5-42

N

named variables

- description 9-16
- examples 9-27—9-33
- mapping 9-65
- naming restrictions 9-16

national language code 5-47

negative sign 4-5

net acks file 5-13

NETPROF 5-4

network

- account number 5-16, 5-28
- acknowledgments
 - continuous receive profile 5-44
 - file 5-13
 - requestor profile 5-14, 5-18
 - trading partner profile 5-23, 5-31
- activity data extract 1-11
- charges 5-18, 5-31
- cmds file 5-21, 5-36
- ID
 - in network profile 5-10
 - in requestor profile 5-16
 - in trading partner profile 5-28
- input file 5-10
- mailbox 5-16
- message class 5-17, 5-30

network *(continued)*

- name 5-10
- output file 5-12
- parameters 5-10
- password 5-17
- program 5-10
- responses file 5-12
- sequence 5-13
- status, updating 10-25
- verifying 5-19

network and trading partner criteria

- date sent 10-11
- internal trading partner ID 10-11
- network acknowledgment date 10-12
- network acknowledgment pending 10-12
- network acknowledgment time 10-12
- network ID 10-11
- network status 10-12
- time sent 10-12
- trading partner nickname 10-11

network profile

- adding a member 5-8
- communication routine 5-10
- description 5-4
- dial connect num 5-13
- input record length 5-10
- members 5-4
- message handler 5-13
- message text header 5-12
- net acks file 5-13
- network ID 5-10
- network input file 5-10
- network name 5-10
- network output file 5-12
- network parameters 5-10
- network program 5-10
- network sequence 5-13
- script name 5-13
- setting up 5-4
- system level 5-12
- system type 5-12
- time zone 5-12
- transaction data queue 5-11
- transaction record length 5-12
- updating a member 5-5

nonrepeating segments 9-39

NOT operator 9-19

NUMBER operator 9-19

numeric data types

- N 6-21, 7-18
- Nn 6-21, 7-19

O

occurrences, specifying during mapping 9-67

operational profiles

- ADAMCTL 4-5
- APPDEFS 4-11
- application definition 4-11
- continuous receive 5-44
- CONTRECV 5-44
- definition of 4-1
- LANGPROF 4-3
- language 4-3
- NETPROF 5-4
- network 5-4
- REQPROF 5-14
- requestor 5-14
- SECUPROF 4-9
- security 4-9
- TPPROF 5-23
- trading partner 5-23
- types 4-2
- user program information 4-5

operators

- arithmetic 9-18
- Boolean 9-18
- comparison 9-18
- order of processing 9-24
- special 9-20
- unary 9-19

Options action 10-34

originator name 4-9

overriding

- application file name 9-95
- application password
 - receive transactions 9-94
 - send transactions 9-83
- application receiver
 - receive transactions 9-90
 - send transactions 9-83
- application sender
 - receive transactions 9-90
 - send transactions 9-83
- envelope profile member 9-83, 9-94
- envelope type 1-7, 9-82
- password
 - receive transactions 9-94
 - send transactions 9-83
- receiver ID
 - receive transactions 9-90
 - send transactions 9-83
- sender ID
 - receive transactions 9-90
 - send transactions 9-83
- trading partner usages
 - receive transactions 9-94
 - send transactions 9-82

P

packed decimal data types

- PD 7-19
- Pn 7-19

page headings, printing on customized documents 14-13

pageable translation 5-48, C-1

panel areas 2-6

panel titles for business documents 13-10

panels

- Add business document (DE03) 13-15
- Add Business Document Definition (DD02) 13-5
- Add Data Element (SC07) 6-9
- Add Data Formats (TD02) 7-7
- Add Data Structure (TD07) 7-12
- Add EDI Standard (SC02) 6-22
- Add Message (DM02) 13-28
- Add Request to Send a File (DF02) 13-35
- Add Segment (SC12) 6-14
- Add Trading Partner Usage for Receiving (TP22) 9-88
- Add Trading Partner Usage for Sending (TP17) 9-78
- Add Transaction (SC18) 6-17
- Add Transaction Mapping (TP02) 9-48
- Add Translation or Validation Table (TM02) 8-6, 8-11
- Add Translation Table (TM09) 8-6
- Add Translation Table Entries (TM10) 8-7
- Add Validation Table (TM03) 8-11
- Add Validation Table Entries (TM04) 8-12
- Additional Record Options (TF53) 10-22
- Additional Selection Criteria (TF08) 10-15
- Administrator's Menu (MP01) 2-5
- Application Criteria (TF06) 10-13
- Archive business document (DE08) 13-20
- Archived Documents (DE21) 13-22
- areas of 2-6
- Associated Objects for Control Strings (EI11) 11-11
- Associated Objects for Document Definitions (EI13) 11-13
- Associated Objects for Document Layouts (EI15) 11-14
- Associated Objects for Standards (EI03) 11-5
- Associated Objects for TP Transactions (EI07) 11-8
- Business Document Definitions (DD01) 13-4, 13-7
- Business Documents (DE01) 13-10
- Confirm Format Change (TD11) 7-7
- Copy Data Format (TD03) 7-7
- Copy Data Structure (TD08) 7-3
- Criteria Selections (TF02) 10-9
- Data Destination (TF51) 10-20
- Data Element Qualifier (TP32) 9-72
- Data Elements (TP09) 9-56
- Data Format Where Used (TD12) 7-7

panels (continued)

- Data Formats (TD01) 7-7
- Data Structures Hierachy (TD06) 7-3
- Delete business document (DE05) 13-18
- Delete Business Document Definition (DD03) 13-7
- Delete Data Format (TD13) 7-7
- Delete File Request (DF03) 13-37
- Delete Message (DM04) 13-31
- Detail and Image Options (TF89) 10-34
- Duplicate Object (EI22) 11-17
- EDI Standards (SC01) 6-15
- Enter Document Key (DE19) 13-14
- Envelope and Send Transactions (TF40) 10-17
- Envelope Criteria (TF05) 10-12
- Envelope Results (TF31) 10-16
- Envelope Transactions (TF30) 10-16
- Event Logging (EL01) 12-4
- Export Application Data Formats (EI05) 11-7
- Export Business Document Definitions (EI12) 11-13
- Export Business Document Layouts (EI14) 11-14
- Export Control Strings (EI10) 11-11
- Export Menu (EI00) 11-4
- Export Profiles (EI23) 11-15
- Export Standards (EI02) 11-5
- Export Tables (EI27) 11-16
- Export TP Transactions (EI06) 11-8
- File business document (DE09) 13-19
- File formatted business document (DE13) 13-21
- File Options (DE15) 13-19
- File Requests (DF01) 13-33
- Functional Acknowledgment Image (TF13) 10-31
- Hierarchical Loop Support (TP40) 9-110
- Import Application Data Formats (EI17) 11-19
- Import Business Document Definitions (EI20) 11-21
- Import Business Document Layouts (EI21) 11-22
- Import Control Strings (EI19) 11-21
- Import Menu (EI01) 11-16
- Import Profiles (EI25) 11-22
- Import Standards (EI16) 11-17
- Import Tables (EI28) 11-24
- Import TP Transactions (EI18) 11-19
- Interactive Entry Facility menu (MM01) 13-3
- Internal Trading Partner ID Selection (DE17) 13-14
- List Libraries (M7100US) 3-4
- List Library Members (M7500US) 3-5
- Log Data (EL04) 12-6
- Loop Segments (TP14) 9-70
- Messages (DM01) 13-27
- Migrate Trading Partner Usage for Receiving (TP38) 9-101
- Migrate Trading Partner Usage for Sending (TP37) 9-100
- Migrate Trading Partner Usages (TP39) 9-102
- Migrate Transaction Mapping (TP36) 9-98
- Network Access (DE14) 13-17
- Network Access (DF08) 13-36

panels (continued)

- Network Access (DM11) 13-30
- Network Access (TF41) 10-17
- Network Acknowledgments Recall (DE16) 13-24
- Network Acknowledgments Recall (DF09) 13-38
- Network Acknowledgments Recall (DM12) 13-32
- Network Acknowledgments Send (DE11) 13-17
- Network Acknowledgments Send (DF06) 13-36
- Network Acknowledgments Send (DM08) 13-17, 13-29
- Network Receive Options (TF61) 10-19
- Print Selections (TF72) 10-35
- Profile Definitions (PM01) 4-1
- Profile Members (EI24) 11-15
- Profile Members (EI26) 11-23
- Qualified Data Elements Mappings (TP15) 9-73
- Qualify Hierarchical Mapping for Receive (TP42) 9-111
- Qualify Hierarchical Mapping for Send (TP41) 9-111
- Qualify Loop Mapping (TP08) 9-69
- Qualify Segment Mapping (TP07) 9-68
- Receive Business Document (DE12) 13-25
- Receive File (DF07) 13-38
- Receive Message (DM10) 13-32
- Receive Transactions (TF60) 10-18
- Related Transactions (TF71) 10-31
- Report Group Status (TF85) 10-26
- Report Selections (TF14) 10-23
- Report Transaction Status (TF10) 10-23
- Retrieve a Library Member (M7800US) 3-5
- Select Business Document Action (DE22) 13-9
- Select Data Format (DE02) 13-13
- Select File Requests Action (DF12) 13-33
- Select Messages Action (DM15) 13-26
- Selection Criteria (EL02) 12-4
- Special Handling for Receiving (TP31) 9-71
- Special Handling for Sending (TP30) 9-60
- Specify Exception and Print Files (TF52) 10-21
- Specify File to Restore (DE20) 13-21
- Standard Transaction Sets (EI04) 11-6
- Status Choices (DE10) 13-16, 13-23
- Status Choices (DF05) 13-35
- Status Choices (DM07) 13-29
- Store Status Criteria (TF07) 10-14
- TP Usage Association (EI30) 11-20
- Trading Partner and Network Criteria (TF04) 10-11
- Trading Partner Selection (DE23) 13-14
- Trading Partner Usage for Receiving (EI09) 11-10
- Trading Partner Usage for Sending (EI08) 11-10
- Trading Partner Usage for Sending (TP16) 9-102
- Trading Partner Usage Overrides for Receiving (TP27) 9-94
- Trading Partner Usage Overrides for Sending (TP26) 9-82
- Transaction Activity Summary (TF11) 10-37

panels *(continued)*

- Transaction Criteria (TF03) 10-9
- Transaction Detail (TF73) 10-31
- Transaction Image (TF75) 10-34
- Transaction Mappings (TP01) 9-48
- Transaction Segments (TP06) 9-52
- Transaction Store Facility Menu (TF01) 10-6
- Transactions (SC17) 6-16
- Translate Received Transactions (TF50) 10-20
- Translation and Validation Tables (TM01) 8-5, 8-10
- Update business document (DE06) 13-22
- Update Business Document Definition (DD04) 13-8
- Update Composite Element (SC32) 6-11
- Update Data Format (TD04) 7-7
- Update Data Structure (TD09) 7-3
- Update Group Status (TF86) 10-27
- Update Interchange Status (TF81) 10-27
- Update Profile Member (PM10) 5-6
- Update Store Status (TF20) 10-25
- Update Translation Table Entries (TM13) 8-8, 8-9
- Update Validation Table Entries (TM07) 8-13
- Utility Command Statements (FF12) 15-3
- Utility Invocation (FF11) 15-2
- View business document (DE07) 13-24
- View Business Document Definition (DD05) 13-8
- View Customized Document (DE18) 13-24
- View Data Format (TD05) 7-7
- View Data Structure (TD10) 7-3
- View File Request (DF04) 13-37
- View Log Entry (EL03) 10-33, 12-6
- View Message (DM06) 13-31
- View Profile Definition (PM05) 4-6
- View Segment (SC15) 6-24

parameters, network 5-10**parent level code** 9-113**password**

- accessing network mailbox 5-17
- application (UNG13) 6-40
- overriding
 - receive transactions 9-94
 - send transactions 9-83
- requestor profile 5-17

password (PW) data type 6-34**point-to-point, connecting** 4-6, 5-4, 5-10, 5-13**post-translate exit routine** 4-5**post-translation exit routine** 9-80**postal code, in TPPROF** 5-30**pre-translate exit routine** 4-5**pre-translation exit routine** 9-91**preparing hierarchical loops** 9-107**preparing the mapping worksheet** 9-43**prerequisites**

- Interactive Entry Facility 13-3
- mapping 9-2

primary environment 5-42**print**

- file
 - document layouts 14-7
 - receiving transactions 10-18
- name 5-46
- type 5-46

Print action 2-11, 10-35**printing**

- business documents 13-22
- customized documents in batch 14-7
- log entries 12-4
- messages 13-31
- page headings on customized documents 14-13
- reports and lists 2-11
- transaction status reports 10-23

profile data, resource names for A-5**profile member** 4-1**profiles**

- activity log 12-1
- ACTLOGS 12-1
- ADAMCTL 4-5
- APPDEFS 4-11
- application definition 4-11
- continuous receive 5-44
- CONTRECV 5-44
- definition of 4-1
- exporting 11-14
- importing 11-22
- LANGPROF 4-3
- language 4-3
- NETPROF 5-4
- network 5-4
- REQPROF 5-14
- requestor 5-14
- SECUPROF 4-9
- security 4-9
- TPPROF 5-23
- trading partner 5-23
- types 4-2
- user program information 4-5

program

- language 4-7
- name 4-6
- name table (PRGMNAME) 8-3

prompt

- application data formats 2-14
- application field name 2-14
- application structure name 2-14
- function key (F4) 2-11
- mapping 2-14

purchase order document layout 14-16**purge**

- date 10-11
- interval 5-48
- pageable 5-48

purge (*continued*)
 track SAP status 5-48
purGe action 10-36

Q

QDATA file 5-11
QDATAE file 5-11
QDATAU file 5-11
qualifying
 data elements 9-71
 loops 9-69
 segments 9-68

R

RACF

 class for DataInterchange A-1
 examples A-11
 general resources services A-13
 profiles, example of defining A-13
 resource names A-8

raw data

 continuous receive profile 5-46
 description 7-1
 record ID 7-13

re-translating received transactions 10-19

real data types

 R 6-21, 7-19
 Rn 6-21, 7-19

recalling from network

 business documents 13-24
 files 13-38
 messages 13-31

receive file name 5-17

receive processing, BIN segment B-7

receiver ID

 ICS07 6-43
 overriding
 receive transactions 9-90
 send transactions 9-83
 qualifier (ICS06) 6-43

receiving

 business documents 13-25
 files 13-38
 free-form messages 5-17
 functional acknowledgments 5-49
 messages 13-32
 new standards 3-7
 nonstandard files 5-17
 transactions 10-17

Recfm keyword B-8

recipient

 name 4-9
 STX05 6-47
 STX06 6-48

recipient (*continued*)

 UNG04 6-39

recipient qualifier (UNG05) 6-39

recipient reference/password (STX10) 6-48

record

 format
 C and D 7-2, 7-5
 raw data 7-1, 7-5
 ID
 data type 7-9
 description 7-13
 length 7-9
 position 7-9
 identifier 7-1
 layout, sample purchase order 9-44

records

 C and D
 description 7-2
 formatting application data for 7-20
 raw data 7-1

reenvolving and sending transactions 10-16

reenvolving transactions 10-15

reference number, interchange 5-36

related element reference numbers 6-12, 6-15

relational definition (Rel Def) 6-12, 6-15

Release action 10-36

release character 5-36

release level (IV) data type 6-34

remote status program 5-22

repeating segments, mapping 9-39

repeating structures in document layouts

Report action 10-36

reporting

 functional group status 10-25
 interchange status 10-25
 transaction status 10-23

reports

 export 11-24
 import 11-24
 status summary 10-24
 transaction status 10-23

REQPROF 5-14

requesting

 functional acknowledgments 5-48
 help 2-9
 new standards 3-3
 standards electronically 3-1

requestor ID 5-16, 5-46

requestor profile

 account number 5-16
 adding a member 5-14
 alt net dial number 5-22
 data line timeout 5-22
 description 5-14
 destination verification 5-19
 EDI option 5-19

requestor profile *(continued)*

- EDI processing override 5-19
- exporting 11-13
- format override 5-20
- message user class 5-17
- network account number 5-16
- network acknowledgment 5-18
- network charges 5-18
- network cmds file 5-21
- network ID 5-16
- network message class 5-17
- password 5-17
- receive file name 5-17
- remote status pgm 5-22
- requestor ID 5-16
- retention period 5-19
- storage format 5-20
- user ID 5-16

requirements designation (Req Des)

- data elements 6-12, 6-14
- segments 6-17

resource names

- Administrator's Menu (MP01) A-3
- application data formats A-6
- archiving log entries A-8
- description A-1
- EDI standards A-6
- envelope standards A-7
- event logs A-5
- export utility A-8
- import utility A-8
- Interactive Entry Facility (IEF) A-4
- migrating mappings A-8
- profile data A-5
- restoring log entries A-8
- trading partner transactions A-6
- Transaction Store Facility A-4
- translation and communications A-7
- translation tables A-6
- validation tables A-6
- where checked A-8

response name 5-47**response type 5-47****responsible agency code 9-90****responsible agency code (GS07) 6-53, 6-59****restoring**

- business documents 13-19
- log entries
 - authority for A-8

restrictions, export/import 11-3**retention period 5-19, 5-32****routing receiver (RR) data type 6-34****routing sender (RS) data type 6-34****S****sample purchase order**

- application data format
 - receive transaction 9-47
 - send transaction 9-46
- EDI format 9-43
- mapping worksheet 9-44
- record layout 9-44
- transaction mapping printout 9-104

SC special operator 9-20**SC01–EDI Standards 6-15****SC02–Add EDI Standard 6-22****SC07–Add Data Element 6-9****SC12–Add Segment 6-14****SC17–Transactions 6-16****SC18–Add Transaction 6-17****SC32–Update Composite Element 6-11****SCH segment 5-35****screens**

- See panels

script name 5-13**scrolling 2-10****SDM LinkPlus 5-13****secondary environment 5-43****SECUPROF 4-9****security**

- administrator A-1
- ID
 - adding a trading partner profile member 5-30
 - security profile (SECUPROF) 4-9
- information (ISA04) 6-55
- information qualifier (ISA03) 6-55
- processes 4-9
- profile
 - member, overriding 9-83
- RACF A-1

security profile (SECUPROF)

- authentication code 4-10
- authentication program 4-10
- authentication type 4-10
- buffer size 4-11
- compression program 4-11
- description 4-9
- encryption program 4-10
- encryption type 4-10
- filtering program 4-11
- filtering type 4-10
- originator name 4-9
- processes 4-9
- recipient name 4-9
- security ID 4-9

segment

- creation for send mapping 9-5
- delimiter 5-35
- ID
 - adding a segment 6-14

segment *(continued)*

ID *(continued)*

- adding a transaction set 6-17
- separator 5-35

segmented output 5-36

segments

- adding 6-13
- characteristics 6-2
- description 6-1
- hierarchical level (HL) 9-106
- mapping 9-3
- maximum use 6-17
- nonrepeating 9-39
- qualifying 9-68
- repeating, in a loop 9-40
- repeating, not in a loop 9-39
- total in interchange
 - EG04 6-52
 - SE01 6-54, 6-60
- total in message
 - MTR01 6-50
 - UNT01 6-41
- total in transaction set (SE01) 6-46
- viewing 6-23

selection criteria 10-7, 10-8

- date and time format 10-9
- default 10-8
- range criteria 10-9
- time and date format 10-9

Send action 10-36

send processing, BIN segment B-2

send/receive program 5-10

sender code (STX03) 6-47

sender ID

- ICS05 6-42
- overriding
 - receive transactions 9-90
 - send transactions 9-83
- qualifier
 - ICS04 6-42
 - UNG03 6-38
- UNG02 6-38

sender name (STX04) 6-47

sending

- business documents 13-17
- files 13-36
- functional acknowledgments 5-48
- messages 13-29

sequence of transfer

- MHD05 6-49
- UNG08 6-41

service segments

- literal keywords 9-33—9-36
- mapping fields for receiving 9-33, 9-34
- mapping fields for sending 9-36

single-byte characters 4-3

Space keyword B-8

special keywords

- &LIT 9-51, 9-52
- &VAR 9-51, 9-52

special operators

- IN 9-20
- IS 9-21
- SC 9-20
- SR 9-21
- TL 9-20
- TO 9-21
- TS 9-20
- UE 9-20

specific name for receiving transactions 10-19

SR special operator 9-21

standalone mode

- description 1-14
- updating profiles 1-16
- updating standards, data formats, or maps 1-17

standard format 10-1

standard ID 6-23

standard transactions, exporting 11-9

standards

- adding EDI
 - description 6-22
 - envelope type 6-23
 - release of a standard 6-23
 - standard ID 6-23
 - standards description 6-23
 - version of a standard 6-23
- customizing
 - EDI 6-4
 - envelope 6-26
- exporting 11-5
- importing 3-1, 11-17
- requesting electronically 3-1
- viewing 6-23
- where to write for information xxii

state, in TPPROF 5-29

status codes

- business document 13-10
- file request 13-34
- message 13-27
- transaction 10-28
- valid store services for 10-29

status summary report 10-24

storage format 5-20, 5-33

storing segments as separate records 5-19

STRUCT statement 14-5

structure (ST) data type 7-19

structure hierarchy 9-38

subelement

- delimiter 5-35
- separator
 - ICS01 6-42
 - ISA16 6-57

- submit date (ICS08)** 6-43
- submit time (ICS09)** 6-43
- substringing data** 9-64
- Summary action** 10-37
- Summary of changes for DataInterchange** xxvii
- SYNCPPOINTS** 5-47
- syntax**
 - conventions in this book xxiv
 - expressions 9-17
 - identifier
 - STX01 6-46
 - UNB01 6-35
 - layout statements 14-2
 - version
 - STX02 6-46
 - UNB02 6-36
- syntax diagram**
 - examples xxiv
 - table xxv
- system ID** 4-14, 5-28
- system level** 5-12
- system profile (SYSPROF)**
 - persistent active 4-14
 - persistent size 4-15
 - persistent threads 4-15
 - system ID 4-14
 - system profile member for CICS 4-14
- system qualifier** 5-28
- system type** 5-12

T

- T envelope profile** 6-46
- TA1 segment** 5-50
- Tab key** 2-9
- tables**
 - alphanumeric table (ALPHANUM) 8-3
 - character set table (CHARSET) 8-3
 - definition of 8-1
 - description 8-4
 - exporting 11-16
 - file name table (FILENAME) 8-3
 - importing 11-23
 - monocase table (MONOCASE) 8-3
 - program name table (PRGMNAME) 8-3
 - suggested uses 8-1
 - supplied by DataInterchange 8-2, 8-9
 - syntax diagram xxv
 - tips 8-1
 - translation 8-4
 - validation 8-9
 - values for identifiers 8-9
- TD01 - Data Formats** 7-7
- TD02 - Add Data Formats** 7-7
- TD03 - Copy Data Format** 7-7

- TD04 - Update Data Format** 7-7
- TD05 - View Data Format** 7-7
- TD06 - Data Structures Hierachy** 7-3
- TD07 - Add Data Structure** 7-12
- TD08 - Copy Data Structure** 7-3
- TD09 - Update Data Structure** 7-3
- TD10 - View Data Structure** 7-3
- TD11 - Confirm Format Change** 7-7
- TD12 - Data Format Where Used** 7-7
- TD13 - Delete Data Format** 7-7
- TDCC standards organization address** xxii
- Technical Information Transaction set (841)** B-1
- test indicator**
 - data type (TI) 6-34
 - ISA15 6-57
 - UNB18 6-38
- test usage**
 - receive transaction 9-93
 - send transaction 9-81
- TF02—Criteria Selections** 10-9
- TF03—Transaction Criteria** 10-9
- TF04—Trading Partner and Network Criteria** 10-11
- TF05—Envelope Criteria** 10-12
- TF06—Application Criteria** 10-13
- TF07—Store Status Criteria** 10-14
- TF08—Additional Selection Criteria** 10-15
- TF10—Report Transaction Status** 10-23
- TF11—Transaction Activity Summary** 10-37
- TF13—Functional Acknowledgment Image** 10-31
- TF14—Report Selections** 10-23
- TF20—Update Store Status** 10-25
- TF30—Envelope Transactions** 10-16
- TF31—Envelope Results** 10-16
- TF40—Envelope and Send Transactions** 10-17
- TF41—Network Access** 10-17
- TF50—Translate Received Transactions** 10-20
- TF51—Data Destination** 10-20
- TF52—Specify Exception and Print Files** 10-21
- TF53—Additional Record Options** 10-22
- TF60—Receive Transactions** 10-18
- TF61—Network Receive Options** 10-19
- TF71—Related Transactions** 10-31
- TF72—Print Selections** 10-35
- TF73—Transaction Detail** 10-31
- TF75—Transaction Image** 10-34
- TF81—Update Interchange Status** 10-27
- TF85—Report Group Status** 10-26
- TF86—Update Group Status** 10-27
- TF89—Detail and Image Options** 10-34
- TIGCMD31** 5-4
- TIGCMV31** 5-4
- TIGD31** 5-4
- TIGV31** 5-4
- time**
 - data type (TM) 6-21, 7-19
 - mask 4-4

time *(continued)*

zone 5-7, 5-12

TL special operator 9-20

TM01–Translation and Validation Tables 8-5, 8-10

TM02–Add Translation or Validation Table 8-6, 8-11

TM03–Add Validation Table 8-11

TM04–Add Validation Table Entries 8-12

TM07–Update Validation Table Entries 8-13

TM09–Add Translation Table 8-6

TM10–Add Translation Table Entries 8-7

TM13–Update Translation Table Entries 8-8, 8-9

TO special operator 9-21

total

segments (TS) data type 6-34

transactions (TT) data type 6-35

TP data phone number 5-36

TP01–Transaction Mappings 9-48

TP02–Add Transaction Mapping 9-48

TP06–Transaction Segments 9-52

TP07–Qualify Segment Mapping 9-68

TP08–Qualify Loop Mapping 9-69

TP09–Data Elements 9-56

TP10–Map Data Element 9-56

TP14–Loop Segments 9-70

TP15–Qualified Data Elements Mappings 9-73

TP16–Trading Partner Usage for Sending 9-102

TP17–Add Trading Partner Usage for Sending 9-78

TP22–Add Trading Partner Usage for

Receiving 9-88

TP26–Trading Partner Usage Overrides for

Sending 9-82

TP27–Trading Partner Usage Overrides for

Receiving 9-94

TP30–Special Handling for Sending 9-60

TP31–Special Handling for Receiving 9-71

TP32–Data Element Qualifier 9-72

TP36–Migrate Transaction Mapping 9-98

TP37–Migrate Trading Partner Usage for

Sending 9-100

TP38–Migrate Trading Partner Usage for

Receiving 9-101

TP39–Migrate Trading Partner Usages 9-102

TP40–Hierarchical Loop Support 9-110

TP41–Qualify Hierarchical Mapping for Send 9-111

TP42–Qualify Hierarchical Mapping for

Receive 9-111

TPPROF 5-23

TRAcKs keyword B-8

trading partner agreements 6-4

trading partner and network criteria

date sent 10-11

internal trading partner ID 10-11

network acknowledgment date 10-12

network acknowledgment pending 10-12

network acknowledgment time 10-12

network ID 10-11

trading partner and network criteria *(continued)*

network status 10-12

time sent 10-12

trading partner nickname 10-11

trading partner capability data extract 1-11

trading partner ID

raw data 7-10

trading partner nickname

continuous receive profile 5-46

trading partner profile 5-28

trading partner profile

data extract 1-11

exporting 11-9, 11-12

trading partner profile (TPPROF)

account number 5-28

adding a member 5-23

address line 2 5-29

address line 3 5-29

allow generic recv 5-37

city 5-29

comment lines 5-36

company address 5-29

company name 5-29

compression 5-37

contact name 5-30

contact phone 5-30

country 5-30

data element delimiter 5-35

data line timeout 5-36

decimal notation 5-35

description 5-23

destination verification 5-32

EDI option 5-32

EDI processing override 5-32

end of text/message 5-34

env profile suffix 5-37

file suffix 5-37

format override 5-33

functional group 5-35

group control number 5-36

interchange control number 5-36

interchange ID 5-29

interchange qualifier 5-29

interchange receive password 5-30

interchange send password 5-30

log standard data 5-35

machine type 5-33

network account number 5-28

network acknowledgment 5-31

network charges 5-31

network cmds file 5-36

network ID 5-28

network message class 5-30

postal code 5-30

release character 5-36

retention period 5-32

trading partner profile (TPPROF) (continued)

- security ID 5-30
- segment delimiter 5-35
- segment ID separator 5-35
- segmented output 5-36
- state 5-29
- storage format 5-33
- subelement delimiter 5-35
- system ID 5-28
- system qualifier 5-28
- TP data phone num 5-36
- trading partner nickname 5-28
- transaction control number 5-36
- user ID 5-29

trading partner transactions

- creating 9-45
- defining 9-47
- description 9-1
- exporting 11-7
- importing 11-19
- resource names A-6

trading partner usages

- adding for sending 9-78
- adding receive transactions 9-88
- exporting 11-9, 11-11
- receive transaction overrides 9-94
- send transaction overrides 9-82

trading partners

- communications overview 5-1
- establishing communications 5-1
- in multiple environments 5-42
- search sequence 5-42
- specifying for maps 9-78
- using interchange ID to determine 5-42

trans rec length 5-7**transaction activity data extract 1-12****transaction code (TC) data type 6-34****transaction criteria**

- date added to store 10-10
- direction 10-10
- duplicate transactions 10-11
- functional acknowledgment date 10-11
- functional acknowledgment pending 10-11
- functional acknowledgment time 10-11
- purge date 10-11
- standard transaction ID 10-10
- test transactions 10-11
- time added to store 10-10
- transaction handle 10-10
- transaction status 10-10
- translation error level 10-10

transaction handle 10-10**transaction mapping printout, sample purchase order 9-104****transaction security profile member name, overriding 9-83****transaction set ID (ST01) 6-45, 6-54, 6-59****transaction sets**

- 841—Technical Information B-1
- adding 6-15
- control number
 - SE02 6-46, 6-54
 - ST02 6-46, 6-54, 6-60
- exporting 11-6
- header and trailer 6-3
- SE02 6-60
- total in interchange
 - EG03 6-52
 - GE01 6-45, 6-53, 6-59
- viewing 6-24

transaction status codes 10-28**Transaction Store**

- data flow
 - receive transactions 10-5
 - send transactions 10-2
- description 10-1

Transaction Store Facility

- authority A-4
- description 10-1

Transaction Store Facility menu (TF01)

- envelope and send transactions option 10-6
- envelope transactions option 10-6
- illustration 10-6
- re-translate received transactions option 10-6
- receive transactions option 10-6
- reenvelope and send transactions option 10-6
- reenvelope transactions option 10-6
- report interchange/group status option 10-7
- report transaction status option 10-6
- translate received transactions option 10-6
- update interchange/group status option 10-7
- update network status option 10-7
- update store status option 10-6

transactions

- authentication key name 9-81
- control number 5-36
- control numbers 5-36, 5-40—5-41
- data queue 5-11
- encryption key name 9-81
- enveloping 10-15
- enveloping and sending 10-16
- limiting selection of 10-7
- managing 10-22
- receiving 10-17
- reenveloping 10-15
- reporting interchange status of 10-25
- reporting status of 10-23
- translating received 10-19
- updating interchange status of 10-27
- updating network status of 10-25

transactions *(continued)*

updating store status of 10-24

Transactions action 10-37

transactions/messages 6-1

Translate action 10-37

translating

business documents 13-26

DataInterchange Utility 15-1

received transactions 10-19

translating application data to application data

(Fixed-to-Fixed) 1-9, 7-23

translation error level 10-10

translation exit routines, exporting 11-9, 11-12

translation resource names A-7

translation tables

adding 8-5

definition of 8-1

description 8-4

exporting 11-9, 11-11

monocase table (MONOCASE) 8-3

resource names A-6

suggested uses

type R tables 8-5

type T tables 8-4

tips 8-1

updating 8-8

transmission control number

BG07 6-51

EG01 6-51

transmission priority code (STX12) 6-49

TS special operator 9-20

turning logging on and off 12-8

Type R (reverse) translation tables 8-5

Type T translation tables 8-4

U

UCount keyword B-8

UCS 999 acknowledgment 5-48

UCS standards organization address xxiii

UCSKEY 6-50

UE special operator 9-20

UNA segment 5-35

unary operator 9-19

unexpected

element check 9-96

segment check 9-96

UNit keyword B-8

Unpurge action 10-37

UNTDIKEY 6-46

updating

business document definitions 13-6

business documents 13-22

composite elements 6-11

envelope standards 6-27

interchange status 10-27

updating *(continued)*

messages 13-30

network profile members 5-5

network status 10-25

network status of transactions 10-25

status of

file requests 13-39

messages 13-32

store status of transactions 10-24

translation tables 8-8

updating functional group status 10-27

validation tables 8-12

usages

See also trading partner usages

receive transactions

active 9-93

test 9-93

send transactions

active 9-81

test 9-81

user exit routines, exporting 11-9, 11-12

user field 5-47

user ID 5-16, 5-29

user program information profile (ADAMCTL)

description 4-5

exit routines 4-5

load module name 4-6

program language 4-7

program name 4-6

programs 4-5

user exits 4-7

utility 15-1

V

validation

data types 9-37

during mapping 9-37

level 9-84, 9-95

translating to application 9-38

translating to standard 9-37

validation tables

adding 8-10

alphanumeric table (ALPHANUM) 8-3

character set table (CHARSET) 8-3

definition of 8-1

description 8-9

exporting 11-6, 11-9, 11-11

file name table (FILENAME) 8-3

program name table (PRGMNAME) 8-3

resource names A-6

suggested uses 8-1

tips 8-1

updating 8-12

variables

DIAPPPFILE 9-24

variables *(continued)*

- DIAPPTYPE 9-24
- DIAUTOCC 9-24
- DICCCTRL 9-24
- DICUSERDATA 9-24
- DIERRFILTER 9-25
- DIEXPTRACE 9-25
- DIMAPCHAIN 9-25
- DIMAPSWITCH 9-25
- DISAPSEQ 9-25
- DIVALLEVEL 9-25
- DIVALTYPE 9-26
- DIVARTRACE 9-26

verifying

- authorization 5-19
- destination 5-19, 5-32

version (VR) data type 6-35

version/release/industry ID (GS08) 6-45, 6-53, 6-59

View Profile Definition (PM05) 4-6

viewing

- business document definitions 13-6
- business documents 13-23
- composite elements 6-25
- data elements 6-25
- file requests 13-37
- log entries 12-4
- messages 13-31
- segments 6-24
- standards 6-23
- transaction sets 6-24

W

windows

- See panels

WINS standards organization address xxiii

X

X envelope profile 6-54

X12

- 999 acknowledgment 5-48
- standards, where to write for information xxii

X12KEY 6-54

Z

zoned decimal data types

- ZD 7-20
- Zn 7-20

Communicating Your Comments to IBM

DataInterchange
Administrator's Guide
Version 3 Release 1

Publication No. SB34-2002-04

If you especially like or dislike anything about this book, please use one of the methods listed below to send your comments to IBM. Whichever method you choose, make sure you send your name, address, and telephone number if you would like a reply.

Feel free to comment on specific errors or omissions, accuracy, organization, subject matter, or completeness of this book. However, the comments you send should pertain to only the information in this manual and the way in which the information is presented. To request additional publications, or to ask questions or make comments about the functions of IBM products or systems, you should talk to your IBM representative or to your IBM authorized remarketer.

When you send comments to IBM, you grant IBM a nonexclusive right to use or distribute your comments in any way it believes appropriate without incurring any obligation to you.

If you are mailing a readers' comment form (RCF) from a country other than the United States, you can give the RCF to the local IBM branch office or IBM representative for postage-paid mailing.

- If you prefer to send comments by mail, use the RCF at the back of this book.
- If you prefer to send comments by FAX, use this number:
United States & Canada: 813+878-5475
- If you prefer to send comments electronically, use this network ID:
IBM Mail Exchange: USIB2PH9 at IBMMAIL

Make sure to include the following in your note:

- Title and publication number of this book
- Page number or topic to which your comment applies.

Readers' Comments — We'd Like to Hear from You

DataInterchange
Administrator's Guide
Version 3 Release 1

Publication No. SB34-2002-04

Overall, how satisfied are you with the information in this book?

	Very Satisfied	Satisfied	Neutral	Dissatisfied	Very Dissatisfied
Overall satisfaction	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

How satisfied are you that the information in this book is:

	Very Satisfied	Satisfied	Neutral	Dissatisfied	Very Dissatisfied
Accurate	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Complete	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Easy to find	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Easy to understand	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Well organized	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Applicable to your tasks	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

Please tell us how we can improve this book:

Thank you for your responses. May we contact you? ☐ Yes ☐ No

When you send comments to IBM, you grant IBM a nonexclusive right to use or distribute your comments in any way it believes appropriate without incurring any obligation to you.

Name

Address

Company or Organization

Phone No.



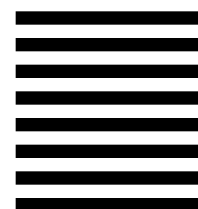
Fold and Tape

Please do not staple

Fold and Tape



NO POSTAGE
NECESSARY
IF MAILED IN THE
UNITED STATES



BUSINESS REPLY MAIL

FIRST-CLASS MAIL PERMIT NO. 40 ARMONK, NEW YORK

POSTAGE WILL BE PAID BY ADDRESSEE

IBM Global Services
Department RD10
PO BOX 20143
TAMPA FL 33633-0872



Fold and Tape

Please do not staple

Fold and Tape



File Number: MVS-79

Program Number: 5685-036 and 5695-076

Printed in the United States of America
on recycled paper containing 10%
recovered post-consumer fiber.

SB34-2002-04

