WebSphere. WebSphere Data Interchange

Version 3.3

IBM

Installation Guide for z/OS

WebSphere® WebSphere Data Interchange

Version 3.3

IBM

Installation Guide for z/OS

SC34-6269-01

> **Note!**
>
> Before using this information and the product it supports, read the information in "Notices" on page 67.

March 2007

This edition applies to IBM WebSphere Data Interchange for MultiPlatforms, V3.3. and to all subsequent releases and modifications until otherwise indicated in new editions.

To send us your comments about this documentation, email doc-comments@us.ibm.com. We look forward to hearing from you.

When you send information to IBM, you grant IBM a nonexclusive right to use or distribute the information in any way it believes appropriate without incurring any obligation to you.

Contents

Tables

About This Book

This book describes how to install Version 3 Release 3 Modification 0 (V3.3) of WebSphere Data Interchange.

Purpose of This Book

This book, in conjunction with the *WebSphere Data Interchange Program Directory* gives instructions on installing WebSphere Data Interchange in a DB2 environment.

Changes after Publication

The *WebSphere Data Interchange Program Directory* document that is shipped with the WebSphere Data Interchange distribution tapes may contain descriptions of the product and current installation information that might have been unavailable when this book was published.

Also check the WebSphere Data Interchange product website for updates to the *WebSphere Data Interchange for z/OS Installation Guide* at the following URL:

http://www.ibm.com/software/integration/wdi/library/pubs/

Audience for This Book

This book is written for the person who installs or maintains WebSphere Data Interchange. You should read one of the following publications before you begin the installation:

* *WebSphere Data Interchange Program Directory*

You should understand the process of installing and maintaining products on your system using System Modification Program Extended (SMP/E), 5668-949.

You should understand how to use DATABASE 2* (DB2*) for z/OS or OS/390, to install the product when WebSphere Data Interchange runs in a DB2 environment.

How This Book Is Organized

Chapter 1, "Installation Configuration Requirements," describes the hardware, software, and storage requirements needed to install and invoke WebSphere Data Interchange.

Chapter 2, "Installing WebSphere Data Interchange," describes the steps required to install WebSphere Data Interchange for all customers using MVS systems and verify that WebSphere Data Interchange is installed properly.

Chapter 3, "Installing WebSphere Data Interchange for CICS Transaction Server for z/OS," describes the steps required to install WebSphere Data Interchange for a CICS Transaction Server for z/OS environment and to verify that WebSphere Data Interchange is installed properly.

Appendix A, "Reorganization of WebSphere Data Interchange Repository Data," describes how to reorganize your WebSphere Data Interchange DB2 tables.

WebSphere Data Interchange and Related Publications

This section describes the documentation available for WebSphere Data Interchange and other publications useful during installtion.

Publications

- *WebSphere Data Interchange for z/OS V3.3 Program Directory* (GI10-2561-02)
- *WebSphere Data Interchange V3.3 Quick Start Guide* (CFOYREN)
- *WebSphere Data Interchange V3.3 Administation and Security Guide* (SC34-6214-01)
- *WebSphere Data Interchange V3.3 User's Guide* (SC34-6215-01)
- *WebSphere Data Interchange V3.3 Messages and Codes* (SC34-6216-01)
- *WebSphere Data Interchange V3.3 Utility Commands and File Formats Reference* (SC23-5873-00)
- *WebSphere Data Interchange V3.3 Programmer's Reference* (SC34-6217-01)
- *WebSphere Data Interchange V3.3 Mapping Guide* (SC23-5874-00)
- *WebSphere Data Interchange V3.3 License File* (GC34-6270-02)
- *IBM SMP/E for z/OS and OS/390 User's Guide* (SA22-7773)
- *IBM SMP/E for z/OS and OS/390 Commands* (SA22-7771)
- *IBM SMP/E for z/OS and OS/390 Reference* (SA22-7772)
- *IBM SMP/E for z/OS and OS/390 Messages, Codes, and Diagnosis* (GA22-7770)

Softcopy Books

All the WebSphere Data Interchange books are available in softcopy format.

Portable Document Format (PDF)

The library is supplied as stand-alone PDFs in US English in the DOC directory on the product CD. The contents of the DOC directory can be viewed without installing the product.

PDF files can be viewed and printed using the Adobe Acrobat Reader. You will need Adobe Acrobat Reader with Search Version 4.05 on Windows NT, or Adobe Acrobat Reader with Search Version 4.5 on UNIX(R) systems.

If you need to obtain the Adobe Acrobat Reader, or would like up-to-date information about the platforms on which the Acrobat Reader is supported, visit the Adobe Systems Inc. Web site at:

`http://www.adobe.com/`

If you cut and paste examples of commands from PDF files to a command line for execution, you must check that the content is correct before you press **Enter**. Some characters might be corrupted by local system and font settings.

Chapter 1. Installation Configuration Requirements

Chapter 1. Installation Configuration Requirements

The WebSphere Data Interchange programs are distributed on standard labeled, 3480 tape cartridge. The *WebSphere Data Interchange Program Directory* documents the contents of the WebSphere Data Interchange distribution tapes.

1.1 Installation Environment

Before you install WebSphere Data Interchange, ensure that you have the proper hardware, software, and storage environments.

1.1.1 Hardware Requirements

The host operating system and the objectives of the installation determine the hardware requirements for WebSphere Data Interchange. The following list describes the minimum system configuration for WebSphere Data Interchange:

- Any hardware capable of running the z/OS (ESA mode) V1.6 and higher operating system.

- A tape drive for installing distributed material.

- A terminal device.

- Between 350 and 400 MB of direct access storage for programs and data

- The hardware required for the network or communication connection.

- A personal computer running the Windows(R) 2000 or Windows(R) XP operating system for running the WebSphere Data Interchange Client.

- LAN network connectivity to support client/server mode for WebSphere Data Interchange Client

1.1.2 Optional Hardware Products

The following list describes the optional hardware.

- The IBM Integrated Cryptographic Feature to support the WebSphere Data Interchange encryption enabling routines running under the IBM Integrated Cryptographic Service Facility/MVS

- The IBM 4753 Network Security Processor to support WebSphere Data Interchange encryption routines running under the IBM 4753 Security Processor MVS Support program

1.1.3 Software Requirements

WebSphere Data Interchange components requires the following operating system:

- z/OS V1R6 or later (5694-A01)

The following list describes the required software:

- DB2 for z/OS or OS/390 V7 (5645-DB2 or later)
- DB2 Connect 8.2
- Windows(R) 2000 or Windows(R) XP for the WebSphere Data Interchange Client
- The software required for the network or communications connection
- IBM System Modification Program Extended (SMP/E) for z/OS V1.04.0 or later (5655-G44)
- IBM XML Toolkit for z/OS, V1.9.   To install this specific version and release of the parser, visit: http://www.ibm.com/servers/eserver/zseries/software/xml

1.1.4 Online response to electronic data interchange requests

This mode requires the CICS(R) environment, which is provided by CICS Transaction Server.   You must purchase and install these products separately. The following levels of CICS are supported:

- CICS Transaction Server for z/OS Version 2.3 (5697-E93) or later
- To perform translation to/from XML formats in CICS Transaction Server, Java SDK 1.4 or later is required. The version of Java you use will depend on which version of CICS Transaction Server you are using.

1.1.5 Optional Software Products

The following list describes the optional software products:

- WebSphere MQ for z/OS and OS/390 V5.3.1 (5655-F10)
- EDI Communication software such as GXS Expedite Base/MVS V4.5 and GXS Expedite/CICS for MVS V4.5
- Software products that support the System Authorization Facility interface, for WebSphere Data Interchange for z/OS security.
- The optional WDI Common Event Handling Java API requires IBM SDK for z/OS, Java 2 Technology Edition, Version 1.4 Sevice Level SR(6) (PTF UK17574) or higher.

1.1.6 Storage Requirements for the Link Pack Area

All WebSphere Data Interchange load modules are reentrant. This means that individual modules can be used simultaneously by two or more tasks. Load modules can also reside in the pageable link pack area (LPA). The modules occupy approximately 23,000 bytes of the pageable LPA, and 6,400,000 bytes of the *extended* pageable LPA. To make the modules LPA-resident, you must copy them into the SYS1.LPALIB data set. You can only load modules from this data set into the LPA.

Each WebSphere Data Interchange user must have at least a 4096 KB region. When you do large amounts of work on the system, you might encounter insufficient storage (TSO region size) errors. The following conditions indicate a memory shortage.

- Request for main storage failed, program terminated

The actual error messages depend on your system. If one of these conditions occur, you need additional storage. The system logs error messages in the Event log file. The Event log file provides details about using the WebSphere Data Interchange event logging option.

1.2 Installation Considerations

Before you install WebSphere Data Interchange, you must be familiar with:

- Repository data
- Program Directory
- National language support
- Security
- Job Control Language
- Data set allocation

1.2.1 Repository Data

WebSphere Data Interchange repository data contains data definitions, formats, and rules that WebSphere Data Interchange uses when processing your data. The installation process loads repository data that WebSphere Data Interchange provides. You can also create repository DB2 data using WebSphere Data Interchange online customization tools.

WebSphere Data Interchange Version 3.3 allows you to choose only a DB2 WebSphere Data Interchange environment.

WebSphere Data Interchange Version 3.3 also provides migration utilities for Version 3.1 and Version 3.2 customers to transfer their existing repository data. You can transfer repository data from DataInterchange Version 3.1 and Version 3.2 DB2 tables to WebSphere Data Interchange Version 3.3 DB2 tables or from DataInterchange Version 3.1 VSAM files to WebSphere Data Interchange Version 3.3 DB2 tables. All installation steps are only for DB2 WebSphere Data Interchange environments.

DataInterchange Version 3.1, WebSphere Data InterchangeVersion 3.2.1, and WebSphere Data Interchange Version 3.3 also provide export/import utilities for customers to transfer all but transaction store and management reporting repository data from Version 3.1 and Version 3.2.1 DB2 tables to Version 3.3 DB2 tables or from Version 3.1 VSAM files to Version 3.3 DB2 tables. For more information about export/import utilities, see *WebSphere Data Interchange V3.3 for Multiplatforms User's Guide*.

If you need further assistance, please call the IBM Support Center. See 1.3, "Targeting Installation Problems" on page 10.

1.2.2 Migration Considerations

For WDI migration considerations, please see 2.1.22, "Step 20. Operational Changes."

1.2.3 Program Directory

Use the *WebSphere Data Interchange Program Directory* with this installation guide. The program directories contain a full description of the WebSphere Data Interchange product, updates to this installation guide, and additional information necessary for installation.

1.2.4 National Language Support

The only language available in this release is English.

Table 1 shows the language identifier, language code, and values for the language that WebSphere Data Interchange currently supports.

Table 1. NLS Language Identifiers and Codes

Language name	ID	Code
US English	ENU	E

1.2.5 Security

Security for WebSphere Data Interchange is provided by Resource Access Control Facility (RACF) or an equivalent product that is consistent with System Authorization Facility (SAF) interfaces.

1.2.6 Job Control Language

You will need to carefully follow the modification directions on all JCL jobs. You will need to define various high-level qualifiers(HLQ), members, dataset names, or DB2 tables. Follow the specific directions for each job carefully analyzing the output for any failures.

1.2.7 Data Set Allocation

You must allocate data sets required for the WebSphere Data Interchange distribution and target libraries during the installation. We recommend that you use separate SMP/E data sets for WebSphere Data Interchange. Comments and instructions presented in this book and the sample JCL assume that you use separate SMP/E data sets. If you use common SMP/E data sets for WebSphere Data Interchange, edit the sample installation JCL in the appropriate steps. The installation requirements section in the *WebSphere Data Interchange Program Directory* provides the information you need to determine the required disk space.

For an example of space allocation for WebSphere Data Interchange DB2 tables, see the *WebSphere Data Interchange Programmer's Reference*. If a reorganization of the WebSphere Data Interchange Version 3.3 DB2 tables is required, see Appendix A, "Reorganization of WebSphere Data Interchange Repository Data" on page 61 for more information.

Table 2 shows the data sets that you might need to allocate. All data set names below can be modified per your specific installation via the appropriate sample installation jobs.

Table 2 (Page 1 of 3). Installation Data Sets

Data Set Name	Function
EDI.V3R3M0.SMPCSI.CSI	Consolidated Software Inventory
EDI.V3R3M0.SMPLOG	History Log Data Set
EDI.V3R3M0.SMPLTS	Load Module Temporary Store
EDI.V3R3M0.SMPMTS	Macro Temporary Store
EDI.V3R3M0.SMPPTS	PTF Temporary Store
EDI.V3R3M0.SMPSCDS	Save Control Data Set
EDI.V3R3M0.SMPSTS	Source Temporary Store
EDI.V3R3M0.SMPLIB1	WebSphere Data Interchange SMP/E Install JCL Library
EDI.V3R3M0.AEDIASM1	WebSphere Data Interchange Assembler Samples Distribution Library
EDI.V3R3M0.AEDICBL1	WebSphere Data Interchange COBOL Samples Distribution Library
EDI.V3R3M0.AEDICCC1	WebSphere Data Interchange C Samples Distribution Library
EDI.V3R3M0.AEDIDAT1	WebSphere Data Interchange Sample Import Data Distribution Library
EDI.V3R3M0.AEDICLS1	WebSphere Data Interchange CLIST Distribution Library
EDI.V3R3M0.AEDIDBR1	WebSphere Data Interchange DB2 DBRM Distribution Library
EDI.V3R3M0.AEDIDB21	WebSphere Data Interchange DB2 Data Distribution Library
EDI.V3R3M0.AEDIINS1	WebSphere Data Interchange Install JCL Distribution Library
EDI.V3R3M0.AEDILMD1	WebSphere Data Interchange Module Distribution Library
EDI.V3R3M0.AEDIPLI1	WebSphere Data Interchange PL/I Samples Distribution Library
EDI.V3R3M0.AEDISAM1	WebSphere Data Interchange Sample JCL Distribution Library
EDI.V3R3M0.AEDISQL1	WebSphere Data Interchange DB2 SQL Distribution Library
EDI.V3R3M0.AEDISTD1	WebSphere Data Interchange DataInterchange Standards Distribution Library
EDI.V3R3M0.AEDIUPD1	WebSphere Data Interchange Future Updates Distribution Library
EDI.V3R3M0.AEDIV3R1	WebSphere Data Interchange Version 3.1 Migration Aid Library
EDI.V3R3M0.AEDIV3R2	WebSphere Data Interchange Version 3.2 Migration Aid Library
EDI.V3R3M0.AEDISTUB	WebSphere Data Interchange Version 3.3 Installation Aid Library

Table 2 (Page 2 of 3). Installation Data Sets

Data Set Name	Function
EDI.V3R3M0.SEDIASM1	WebSphere Data Interchange Assembler Samples Target Library
EDI.V3R3M0.SEDICBL1	WebSphere Data Interchange COBOL Samples Target Library
EDI.V3R3M0.SEDICCC1	WebSphere Data Interchange C Samples Target Library
EDI.V3R3M0.SEDICLS1	WebSphere Data Interchange CLIST Target Library
EDI.V3R3M0.SEDIDAT1	WebSphere Data Interchange Sample Import Data Target Library
EDI.V3R3M0.SEDIDBR1	WebSphere Data Interchange DB2 DBRM Target Library
EDI.V3R3M0.SEDIDB21	WebSphere Data Interchange DB2 Data Target Library
EDI.V3R3M0.SEDIINS1	WebSphere Data Interchange Install JCL Target Library
EDI.V3R3M0.SEDILMD1	WebSphere Data Interchange Load Module Target Library
EDI.V3R3M0.SEDIPLI1	WebSphere Data Interchange PL/I Samples Target Library
EDI.V3R3M0.SEDISAM1	WebSphere Data Interchange Sample JCL Target Library
EDI.V3R3M0.SEDISQL1	WebSphere Data Interchange DB2 SQL Target Library
EDI.V3R3M0.SEDISTD1	WebSphere Data Interchange DataInterchange Standards Distribution Library
EDI.V3R3M0.SEDIUPD1	WebSphere Data Interchange Future Updates Target Library
EDI.V3R3M0.SEDIV3R1	WebSphere Data Interchange Version 3.1 Migration Aid Library
EDI.V3R3M0.SEDIV3R2	WebSphere Data Interchange Version 3.2 Migration Aid Library
EDI.V3R3M0.SEDISTUB	WebSphere Data Interchange Version 3.3 Installation Aid Library
EDI.V3R3M0.AEDIASM2	WebSphere Data Interchange Assembler Samples Distribution Library
EDI.V3R3M0.AEDICBL2	WebSphere Data Interchange COBOL Samples Distribution Library
EDI.V3R3M0.AEDICCC2	WebSphere Data Interchange C Samples Distribution Library
EDI.V3R3M0.AEDIDBR2	WebSphere Data Interchange DB2 DBRM Distribution Library
EDI.V3R3M0.AEDIINS2	WebSphere Data Interchange Install JCL Distribution Library
EDI.V3R3M0.AEDIHFS2	WebSphere Data Interchange XML Java Distribution Library
EDI.V3R3M0.AEDILMD2	WebSphere Data Interchange Module Distribution Library
EDI.V3R3M0.AEDIPLI2	WebSphere Data Interchange PL/I Samples Distribution Library
EDI.V3R3M0.AEDISAM2	WebSphere Data Interchange Sample JCL Distribution Library
EDI.V3R3M0.AEDISQL2	WebSphere Data Interchange DB2 SQL Distribution Library
EDI.V3R3M0.AEDIUPD2	WebSphere Data Interchange Future Updates Distribution Library
EDI.V3R3M0.SEDIASM2	WebSphere Data Interchange Assembler Samples Target Library
EDI.V3R3M0.SEDICBL2	WebSphere Data Interchange COBOL Samples Target Library
EDI.V3R3M0.SEDICCC2	WebSphere Data Interchange C Samples Target Library

Table 2 (Page 3 of 3). Installation Data Sets

Data Set Name	Function
EDI.V3R3M0.SEDIDBR2	WebSphere Data Interchange DB2 DBRM Target Library
EDI.V3R3M0.SEDIINS2	WebSphere Data Interchange Install JCL Target Library
EDI.V3R3M0.SEDIHFS2	WebSphere Data Interchange XML Java Target Library
EDI.V3R3M0.SEDILMD2	WebSphere Data Interchange Load Module Target Library
EDI.V3R3M0.SEDIPLI2	WebSphere Data Interchange PL/I Samples Target Library
EDI.V3R3M0.SEDISAM2	WebSphere Data Interchange Sample JCL Target Library
EDI.V3R3M0.SEDISQL2	WebSphere Data Interchange DB2 SQL Target Library
EDI.V3R3M0.SEDIUPD2	WebSphere Data Interchange Future Updates Target Library

1.3 Targeting Installation Problems

If an error occurs, use the error message number to refer to the appropriate book for further information.

If you cannot resolve your problem:

- Call the IBM Support Center toll free at 1-800-IBM-SERV (1-800-426-7378) if you are in the continental United States, Alaska, Puerto Rico, Hawaii, or Guam.
- Go to the following web site:

  `http://techsupport.services.ibm.com/guides/contacts.html`

  From the **Support Contacts** list, select your country for your support contact information.

Submit any required APARs to the location responsible for the failing component.

This table identifies the component ID (COMP ID) and the field engineering service number (FESN) for WebSphere Data Interchange.

Table 3. Component ID and Field Engineering Service Number

COMP ID	Component name	FESN
5655I4003	WebSphere Data Interchange for z/OS	0500287

1.4 WebSphere Data Interchange Installation Paths

The following tables define the path requirements, in step order, for each installation procedure. You must clearly define installation considerations (on page 5) before you begin the installation procedure.

Table 4 describes which table in this section applies to your installation.

Table 4. WebSphere Data Interchange Installation Paths

Installing WebSphere Data Interchange as new customer	See Table 5 on page 11
Upgrading from DataInterchange/MVS Version 3.1 or from WebSphere Data Interchange Version 3.2 to Version 3.3	See Table 6 on page 13
Installing WebSphere Data Interchange for CICS	See Table 7 on page 15

1.4.1 Installing WebSphere Data Interchange for z/OS for a New Customer

Follow these steps only if you are a new WebSphere Data Interchange for z/OS customer using a DB2 environment.

Table 5 (Page 1 of 2). Installing WebSphere Data Interchange for z/OS (DB2)

Step	Description	Member Name	Page
1	Copying sample installation JCL library	EDIJCOPY	20
2	SMP/E Considerations for Installing WebSphere Data Interchange		22
3	Allocating SMP/E control data sets (Optional Job)	EDIJASMP	23
4	Initializing the SMP/E CSI data set (Optional Job)	EDIJINIT	23
5	Allocating WebSphere Data Interchange target and distribution libraries	EDIALLOC	24
6	WebSphere Data Interchange SMP/E DDDEFs	EDIDDDEF	24
7	WebSphere Data Interchange SMP/E RECEIVE	EDIRECEV	24
8	WebSphere Data Interchange SMP/E APPLY	EDIAPPLY	25
9	WebSphere Data Interchange SMP/E ACCEPT	EDIACCEP	26
10	Processing Cumulative Service Tape, if Applicable		26
11	Creating Version 3.3 repository DB2 objects		27
12	Loading repository DB2 tables	EDIJRDLD	29
13	Running RUNSTATS on the loaded DB2 tables	EDIJRNST	30
14	Binding the DB2 plan	EDIJBIND	30
15	Granting authorization on the DB2 objects	EDISGRNT	30
16	Importing sample default data	EDIJDAT1	32
19	Verifying the installation		35

Table 5 (Page 2 of 2). Installing WebSphere Data Interchange for z/OS (DB2)

Step	Description	Member Name	Page
22	Common Event Handler (optional)		38

1.4.2 Upgrading from DataInterchange/MVS Version 3.1 or from WebSphere Data Interchange Version 3.2 to Version 3.3

Follow these steps only if you are upgrading DataInterchange/MVS Version 3.1 to Version 3.3 or DataInterchange/MVS Version 3.2 to Version 3.3 in a DB2 environment.

Customers should be at PTF level UK10658, dated Jan 10, 2006. This will mean that all database maintenance will have been applied to the WDI V3.2 system. Please make sure you are at the required PTF level.

Table 6 (Page 1 of 2). Upgrading DataInterchange/MVS from Version 3.1 or Version 3.2 to Version 3.3 (DB2)

Step	Description	Member Name	Page
1	Copying sample installation JCL library	EDIJCOPY	20
2	SMP/E Considerations for Installing WebSphere Data Interchange		22
5	Allocating WebSphere Data Interchange target and distribution libraries	EDIALLOC	24
6	WebSphere Data Interchange SMP/E DDDEFs	EDIDDDEF	24
7	WebSphere Data Interchange SMP/E RECEIVE	EDIRECEV	24
8	WeSphere Data Interchange SMP/E APPLY	EDIAPPLY	25
9	WebSphere Data Interchange ACCEPT	EDIACCEP	26
10	Processing Cumulative Service Tape, if Applicable		26
11	Creating Version 3.3 repository DB2 objects		27
12	Loading repository DB2 tables	EDIJRDLD	29
13	Running RUNSTATS on the loaded DB2 tables	EDIJRNST	30
14	Binding the DB2 plan	EDIJBIND	30
15	Granting authorization on the DB2 objects	EDISGRNT	30
16	Importing sample default data	EDIJDAT1	32
17a	(**If migrating from 3.1 or 3.2 DB2**) Build Control file for exporting Version 3.1 or 3.2 DB2 Data	EDIJXPAD	32

Step	Description	Member Name	Page
17b	(**If migrating from 3.1 VSAM**) Build Control file for exporting Version 3.1 non-DB2 Data	EDIJXPAV	33
18a	(**If Migrating from 3.1 or 3.2 DB2**) Migrate Version 3.1 or Version 3.2 DB2 data to Version 3.3	EDIJEXPD EDIJRNST	33
18b	(**If Migrating from 3.1 VSAM**) Migrate Version 3.1 non-DB2 data to DB2 environment	EDIJEXVD EDIJRNST	34
19	Verifying the installation		35
20	Reviewing operational enhancements		36
21	Migrate operational data (Transaction store, Management reporting, SAP status) from WDI V3.2	See APPENDIX B	63
22	Common Event Handler (Optional)		38

1.4.3 Installing WebSphere Data Interchange for CICS

Note: You must complete the installation steps in Table 5 on page 11 before installing the WebSphere Data Interchange CICS portion.

Customers migrating from a previous WDI release be aware that these instructions assume you will be defining WDI 3.3 in a new CICS region. In other words, all RDO definitions herein use group name, EDIGROUP, which is the same as prior WDI releases. Therefore, if you are using a CSD from a prior WDI release, then either change the EDIGROUP group name throughout this installation, or delete the existing EDIGROUP from the CSD before proceeding. Then, after completing the RDO definitions below, add the chosen group name to your list.

Table 7. Installing WebSphere Data Interchange for CICS

Step	Description	Member Name	Page
23	Creating COBOL user exit enabling module for CICS	FXXJLCBL FXXJLCB2	45
24	Creating the EDIW transaction module for CICS	FXXJEDIW	46
25	Creating DB2 synonyms for CICS	FXXSSYNO	46
26	Binding the DB2 plan	FXXJBIND	47
27	Granting authorization on the DB2 objects		47
28	Setting up DCT for WebSphere Data Interchange	FXXODCT	48
29	Setting up PCT for WebSphere Data Interchange	FXXOPCT	48
30	Setting up PPT for WebSphere Data Interchange	FXXOPPT	49
31	Setting up RCT	FXXORCT	49
32	PLT considerations		49
33	Setting up XLT for WebSphere Data Interchange and GXS Expedite/CICS	FXXOXLT	51
34	XML setup considerations		51
35	Setting Up Error Notification Support for WebSphere Data Interchange		51
36	Customizing the CICS startup JCL for DB2 WebSphere Data Interchange	FXXDDND	56
37	Verifying the CICS installation	EDIV	57

Chapter 2. Installing WebSphere Data Interchange

Chapter 2. Installing WebSphere Data Interchange

This chapter describes the steps required to install WebSphere Data Interchange for z/OS and verify that the installation was successful. **CICS Transaction Server for z/OS** customers must complete the installation steps in this chapter before installing the CICS portion.

This chapter also provides information about:

- Setting up System Modification Program/Extended (SMP/E) libraries for new and existing customers on z/OS systems. Version 3.3 is installed for new systems and existing systems are upgraded from Version 3.1 and Version 3.2 to Version 3.3

- Installing DB2 environments.

- Verifying the completed z/OS installation.

2.1 Installation Steps

This process installs WebSphere Data Interchange using the following naming conventions:

- EDI is the default HLQ (high-level qualifier) of each SMP/E data set. The sample JCL supplied on the distribution tape follows this convention. This can be modified for your installation. See the appropriate sample job steps to modify.

- ENU is the default three-character language identifier.

- EDIENU33 is the high-level qualifier of each WebSphere Data Interchange DB2 table.

2.1.1 Step 1. Copying Sample Installation JCL Library

EDIJCOPY is included on the WebSphere Data Interchange tape cartridge and provides a set of sample job streams for product installation. EDIJCOPY uses the IBM IEBCOPY utility program to copy the sample job streams to a new data set called EDI.V3R3M0.SMPLIB1. After applying any maintenance (step 10 - Cumulative Service Tape), you should use the sample job streams in the target library EDI.V3R3M0.SEDIINS1.

You may also access the sample installation jobs by performing an SMP/E RECEIVE for the FMID H28N330 for WebSphere Data Interchange and then copying the jobs from dataset **EDI.TLIB.H28N330.F2** which represents your RECEIVEd RELFILE F2.

The following steps describe how to copy the JCL library sample installation files.

__ 1. Create the following job stream. Use your system editor to create the JCL.

```
//EDIJCOPY JOB (INSTALLATION DEPENDENCIES)
//*
//**********************************************************
//*  COPY WEBSPHERE DATA INTERCHANGE SAMPLE SMP INSTALL    *
//*       JCL MEMBERS                                       *
//**********************************************************
//*                                                        *
//*    WEBSPHERE DATA INTERCHANGE VERSION 3 RELEASE 3      *
//*                                                        *
//*    5655-I40 COPYRIGHT IBM CORP. 1997, 2007             *
//*    ALL RIGHTS RESERVED.                                *
//*    US GOVERNMENT USERS RESTRICTED RIGHTS -             *
//*    USE, DUPLICATION OR DISCLOSURE RESTRICTED BY GSA ADP*
//*    SCHEDULE CONTRACT WITH IBM CORP.                    *
//*                                                        *
//**********************************************************
//*  1. CHANGE THE JOB STATEMENT AS NECESSARY              *
//*  2. IF NECESSARY CHANGE:                               *
//*     _"TAPUNIT" PARAMETER VALUE                         *
//*     _"HLQ"     PARAMETER VALUE                         *
//*     _"RELS"    PARAMETER VALUE                         *
//*     _"UNIT"    PARAMETER VALUE                         *
//*     _"OUTC"    PARAMETER VALUE                         *
//**********************************************************
//*
//EDIJCOPY PROC EDIFMID='H28N330',
//            TAPVOL='28N330',
//            TAPUNIT='3480',
//            HLQ='EDI',
//            RELS='V3R3M0',
//            UNIT='SYSALLDA',
//            OUTC='*'
//*
//STEP1    EXEC PGM=IEBCOPY
//SYSPRINT DD SYSOUT=&OUTC
//SYSUT3   DD UNIT=&UNIT,SPACE=(80,(15,1))
//TAPEIN   DD DSN=IBM.&EDIFMID..F2,UNIT=&TAPUNIT,
//            VOL=SER=&TAPVOL,LABEL=(3,SL),DISP=OLD
//DISKOUT  DD DSN=&HLQ..&RELS..SMPLIB1,
//            SPACE=(0,(224,56,10)),DISP=(NEW,CATLG,DELETE),
//            UNIT=&UNIT,DCB=(RECFM=FB,LRECL=80,BLKSIZE=0)
// PEND
//*
//       EXEC EDIJCOPY
//STEP1.SYSIN DD *
   COPY INDD=TAPEIN,OUTDD=DISKOUT
/*
//
```

__ 2. Modify the parameters for your installation.

__ 3. Run the job.

__ 4. Verify that all return codes are zero. If you receive a nonzero return code, see the appropriate
product's message manual for further information.

__ 5. If all return codes are zero, go to 2.1.2, "Step 2. SMP/E Considerations for Installing WebSphere
Data Interchange."

2.1.2 Step 2. SMP/E Considerations for Installing WebSphere Data Interchange

This release of WebSphere Data Interchange is installed using the SMP/E RECEIVE, APPLY, and ACCEPT commands. The SMP/E dialogs may be used to accomplish the SMP/E installation steps.

All SMP/E installation jobs provided assume that all necessary DD statements for the execution of SMP/E are defined using DDDEFs. Sample jobs are provided to assist you in installing WebSphere Data Interchange.

The recommended values for some SMP/E CSI subentries are shown in the following table. Use of values lower than these may result in failures in the installation process. DSSPACE is a subentry in the GLOBAL options entry. PEMAX is a subentry of the GENERAL entry in the GLOBAL options entry. Refer to the *SMP/E Reference* for instructions on updating the global zone.

Table 8. SMP/E Options Subentry Values

SUB-ENTRY	Value	Comment
DSSPACE	(2500,500,175)	
PEMAX	SMP/E Default	IBM recommends using the SMP/E default for PEMAX

WebSphere Data Interchange uses the CALLLIBS function provided in SMP/E to resolve the Encryption/Decryption/Authentication routines, TSO Service routine, and C run-time environment subroutines during installation. The following CALLLIBS are used during the APPLY of WebSphere Data Interchange. The libraries are used to resolve external references during the link-edit and no changes are made to the library.   The LPALIB, SCEELKED, CSSLIB, and SDFHLOAD CALLLIBS are **REQUIRED** for WebSphere Data Interchange.   The other CALLLIBS are optional.   If you do not have or need the optional CALLLIBS, you must point your DDDEF statement to the WebSphere Data Interchange's STUB library, EDI.V3R3M0.SEDISTUB. If you do not do this, your SMP/E APPLY will fail. Complete directions can be found in the EDIDDDEF sample job.

- LPALIB
- SCEELKED
- CSSLIB
- SDFHLOAD
- SDSNLOAD
- SCSQLOAD
- TCSULIB
- TCSUOBJ

Note: For WebSphere Data Interchange, 200 tracks of 3380/3390 DASD space is adequate for the SMPLTS. Refer to *SMP/E Reference* for additional information on the SMPLTS data set.

Go to 2.1.3, "Step 3. Allocating SMP/E Control Data Sets (Optional Job)." if you are creating separate or new SMP/E CSI data sets for WebSphere Data Interchange V3.3.   If not, go to 2.1.5, "Step 5. Allocating WebSphere Data Interchange Target and Distribution Libraries." to install into your SMP/E CSI.

2.1.3 Step 3. Allocating SMP/E Control Data Sets (Optional Job)

Note: If you do not use separate SMP/E data sets for WebSphere Data Interchange, go to 2.1.5, "Step 5. Allocating WebSphere Data Interchange Target and Distribution Libraries." This is an optional job only if you wish to create a new SMP/E CSI using the sample EDIJASMP job.

EDIJASMP runs the IBM IEFBR14 utility program to allocate space for all SMP/E control data sets. EDIJASMP also uses the IBM Access Method Services program (IDCAMS) to initialize CSI areas. The installation uses separate SMP/E data sets for WebSphere Data Interchange installation.

The following steps describe how to allocate SMP/E control data sets.

__ 1. Edit the EDIJASMP member in EDI.V3R3M0.SMPLIB1.

   Modify the member for your environment as shown in the comment at the beginning of the JCL.

__ 2. Run the job.

__ 3. Verify that all return codes are zero, and that all data sets were allocated correctly. If you receive a nonzero return code, see the appropriate product's message manual for further information.

__ 4. If all return codes are zero, go to 2.1.4, "Step 4. Initializing the SMP/E CSI Data Set (Optional Job)."

2.1.4 Step 4. Initializing the SMP/E CSI Data Set (Optional Job)

EDIJINIT places global, target, and distribution zone information in the System Modification Program/Extended Consolidated Software Inventory (SMP/E CSI) data set. You will need to define your TARGET and DISTRIBUTION names are well as modify any OPTIONS information. This is an optional job only needed if you ran the EDIJSAMP job previously.

The following steps describe how to initialize the SMP/E CSI data sets.

__ 1. Edit the EDIJINIT member in EDI.V3R3M0.SMPLIB1.

   Modify the member for your environment as shown in the comment at the beginning of the JCL.

__ 2. Run the job.

__ 3. Verify that all return codes are zero. If you receive a nonzero return code, see the appropriate product's message manual for further information.

__ 4. If all return codes are zero, go to 2.1.5, "Step 5. Allocating WebSphere Data Interchange Target and Distribution Libraries."

2.1.5 Step 5. Allocating WebSphere Data Interchange Target and Distribution Libraries

EDIALLOC runs the IBM IEFBR14 utility program to allocate space for all WebSphere Data Interchange distribution and target libraries.

Note: If you plan to maintain both WebSphere Data Interchange Version 3.3 and the previous version, use separate SMP/E data sets for WebSphere Data Interchange Version 3.3.

The following steps describe how to allocate WebSphere Data Interchange target and distribution libraries.

__ 1. Edit the EDIALLOC member in EDI.V3R3M0.SMPLIB1.

   Modify the member for your environment as shown in the comment at the beginning of the JCL.

__ 2. Run the job.

__ 3. Verify that all return codes are zero, and that you correctly allocated all data sets. If you receive a nonzero return code, see the appropriate product's message manual for further information.

__ 4. If all return codes are zero, go to 2.1.6, "Step 6. WebSphere Data Interchange DDDDEFs."

2.1.6 Step 6. WebSphere Data Interchange DDDDEFs

EDIDDDEF updates the global, target, and distribution zone information in your SMP/E CSI data set for WebSphere Data Interchange.

The following steps describe how to update the SMP/E data sets.

__ 1. Edit the EDIDDDEF member in EDI.V3R3M0.SMPLIB1.

   Modify the member for your environment as shown in the comment at the beginning of the JCL.

__ 2. Run the job.

__ 3. Verify that all return codes are not greater than 0004, and that data set allocations were updated correctly. If you receive a return code greater than 0004, see the appropriate product's message manual for further information.

__ 4. If all return codes are not greater than 0004, go to 2.1.7, "Step 7. WebSphere Data Interchange RECEIVE."

2.1.7 Step 7. WebSphere Data Interchange RECEIVE

EDIRECEV issues the SMP/E RECEIVE command to store WebSphere Data Interchange system modifications (SYSMODs) in the PTS control data set. EDIRECEV also records SYSMOD and exception data in the consolidated software inventory (CSI) global zone.

The following steps describe how to process the SMP/E RECEIVE.

__ 1. Edit the EDIRECEV member in EDI.V3R3M0.SMPLIB1.

Modify the member for your environment as shown in the comment at the beginning of the JCL.

Pay special attention to the the OPTIONS(#optionsname) parameter on the SET BDY(GLOBAL) OPTIONS(#optionsname) statement.   Change '#optionsname' to the name of your SMP/E OPTIONS entry used when initializing your SMP/E CSI.   If you used the EDIJINIT JCL job from Step 4, this entry would be OPTIONS(EDIOPT).   If this does not apply to your installation, delete the OPTIONS entry parameter on the SET BDY(GLOBAL) line to:

```
SET BDY(GLOBAL)  .
```

__ 2. Run the job.

__ 3. Verify that all return codes are zero. If you receive a nonzero return code, see the appropriate product's message manual for further information.

__ 4. If all return codes are zero, go to 2.1.8, "Step 8. WebSphere Data Interchange APPLY."

2.1.8 Step 8. WebSphere Data Interchange APPLY

To receive the full benefit of the SMP/E Causer SYSMOD Summary Report, do *not* bypass the following on the APPLY CHECK: PRE, ID, REQ, and IFREQ. This is because the SMP/E root cause analysis identifies the cause only of **ERRORS** and not of **WARNINGS** (SYSMODs that are bypassed are treated as warnings, not errors, by SMP/E).

The following steps describe how to process SMP/E APPLY.

__ 1. Edit the EDIAPPLY member in EDI.V3R3M0.SMPLIB1.

Modify the member for your environment as shown in the comment at the beginning of the JCL.

__ 2. Run the job.

__ 3. If you receive a return code greater than 0000, see the appropriate product's message manual for further information.

__ 4. Once you have taken any actions indicated by the APPLY CHECK, edit the EDIAPPLY member in EDI.V3R3M0.SMPLIB1, remove the CHECK operand and run the job again to perform the APPLY.

Note: The GROUPEXTEND operand indicates that SMP/E apply all requisite SYSMODs. The requisite SYSMODS might be applicable to other functions.

__ 5. Verify that the return code is 0004. If you receive a return code greater than 0004, see the appropriate product's message manual for further information.

Note: If you rerun the job, you must insert the REDO operand in the APPLY command.

__ 6. If the return code is 0004, go to 2.1.9, "Step 9. WebSphere Data Interchange SMP/E ACCEPT."

2.1.9 Step 9. WebSphere Data Interchange SMP/E ACCEPT

To receive the full benefit of the SMP/E Causer SYSMOD Summary Report, do *not* bypass the following on the ACCEPT CHECK: PRE, ID, REQ, and IFREQ. This is because the SMP/E root cause analysis identifies the cause only of **ERRORS** and not of **WARNINGS** (SYSMODs that are bypassed are treated as warnings, not errors, by SMP/E).

Before using SMP/E to load new distribution libraries, it is recommended that you set the ACCJCLIN indicator in the distribution zone. This will cause entries produced from JCLIN to be saved in the distribution zone whenever a SYSMOD containing inline JCLIN is ACCEPTed. For more information on the ACCJCLIN indicator, see the description of inline JCLIN in the SMP/E manuals.

The following steps describe how to process SMP/E ACCEPT.

__ 1. Edit the EDIACCEP member in EDI.V3R3M0.SMPLIB1.

Modify the member for your environment as shown in the comment at the beginning of the JCL.

__ 2. Run the job.

__ 3. Verify that all return codes are 0000. If you receive a return code greater than 0000, see the appropriate product's message manual for further information.

Once you have taken any actions indicated by the ACCEPT CHECK, remove the CHECK operand and run the job again to perform the ACCEPT.

Note: The GROUPEXTEND operand indicates that SMP/E accept all requisite SYSMODs. The requisite SYSMODS might be applicable to other functions.

__ 4. Verify that all return codes are not greater than 0004. If you receive a return code greater than 0004, see the appropriate product's message manual for further information.

Note: If you rerun the job, you must insert the REDO operand in the ACCEPT command.

__ 5. If the return code is not greater than 0004, go to 2.1.10, "Step 10. Processing Cumulative Service Tape, if Applicable."

2.1.10 Step 10. Processing Cumulative Service Tape, if Applicable

A cumulative service tape may be shipped with the product package. If no cumulative service tape is shipped or if the product is received by CBPDO, skip this section.

If a cumulative service tape was received with the product package, RECEIVE the PTFs and HOLDDATA from the cumulative service tape. APPLY the service after RECEIVE processing has completed. It is not necessary to execute PTF HOLD ACTIONs because the remaining install process will automatically incorporate any changes.

After RECEIVE / APPLY processing for the cumulative service tape has been completed, do one of the following:

  • If all return codes are zero, go to 2.1.11, "Step 11. Creating Version 3.3 Repository DB2 Objects."

2.1.11 Step 11. Creating Version 3.3 Repository DB2 Objects

2.1.11.1 Notes For All Customers

ENU is the default three-character language identifier, and E is the default one-character language code. If necessary, you may change the default language identifier ENU for your installation.

When you install WebSphere Data Interchange you should:

- All SQL statements supplied on the distribution tape create each WebSphere Data Interchange table, view, index, etc. with 'EDIENU33' as the high-level qualifier. In which case, you must define a TSO ID 'EDIENU33' and give it DB2 SYSADM authority, then use this id to create all objects. Use 'SET CURRENT SQLID=EDIENU33' to switch to this ID before creating all of the objects.

- You can use a high-level-qualifier value other than the provided 'EDIENU33'. In which case, this other value must match the name of the ID creating all objects. Use 'SET CURRENT SQLID=xxxxxxxx' to switch to the ID.

- If the SQLID creating the objects does not match the high-level qualifier, then you will receive various DSNX200I messages during the EDIJBIND job.

- You may need to increase the DB2 sort area used for 32K page sizes. Contact your DBA for assistance with this. A size of 1000 cylinders is recommended, although the actual size required may be more or less depending on the types of Document Store queries that you do and the amount of data in your Document Store tables. Having a sort area that is too small may result in DB2 errors for certain types of Document Store queries.

- The EDIDB33E database is created with CCSID UNICODE. DB2 can use the conversion services of z/OS support for Unicode only when the conversion environment is active. Be sure this environment is active and supports the needed code page conversions for your environment. For more information, please refer to the DB2 Installation Guide, *Customizing support for Unicode*. If inactive, you will encounter errors when executing various WDI functions. For example:

```
DSNT408I SQLCODE = -332, ERROR:  CHARACTER CONVERSION
BETWEEN CCSID 00367 TO 01208 REQUESTED BY DSNXROUA IS NOT SUPPORTED
```

  To correct this error, you would add CONVERSION 0367,1208 to your conversion environment. A helpful system command "DISPLAY UNI,ALL" displays available conversions on your system.

2.1.11.2 Notes For Customers Migrating From A Previous Release

The installation procedures in this section preserve data that exists in your current environment. This section describes jobs that migrate existing Version 3.1 or Version 3.2 repository data into the new Version 3.3 tables. The migration process copies the Version 3.1 or Version 3.2 data into temporary files. When you finish the installation, you may delete the temporary files.

Customers should be at PTF level UK10658, dated Jan 10, 2006. This will mean that all database maintenance will have been applied to the WDI V3.2 system. Please make sure you are at the required PTF level.

Note: If you plan to maintain your previous version of DataInterchange separately from your WebSphere Data Interchange Version 3.3 system, use separate SMP/E data sets for WebSphere Data Interchange Version 3.3. For SMP/E data set allocation and initialization, see 2.1.3, "Step 3. Allocating SMP/E Control Data Sets (Optional Job)" on page 23 and 2.1.4, "Step 4. Initializing the SMP/E CSI Data Set (Optional Job)" on page 23.

YOUR PREVIOUS RELEASE MUST BE EXECUTABLE FOR THE DURATION OF THIS MIGRATION.

If you are installing Version 3.3 on the same DB2 subsystem as your previous V3.1 or V3.2 system, you must use a different SQLID. Additionally, the plan names must be unique.

Note: If you are migrating from Version 3.1, this migration does **NOT** preserve transaction store, management reporting, SAP status. Event log data is not migrated from any previous releases into the Version 3.3 database.

Use the DB2 Structured Query Language (SQL) Processor Using File Input (SPUFI) facility under Interactive System Productivity Facility/Program Development Facility (ISPF/PDF) to create WebSphere Data Interchange Version 3.3 repository DB2 objects. Members in the distributed data set, EDI.V3R3M0.SEDISQL1, contain the SQL statements.

The following steps describe how to access SQL statements and submit them to DB2 for processing using the DB2 SPUFI option under ISPF/PDF.

__ 1. On the DB2I Primary Option Menu, select the SPUFI option (normally option 1).

__ 2. On the SPUFI panel, Enter the name **EDI.V3R3M0.SEDISQL1(EDISDB2)** to update the SQL statements.

- For heavily updated tables, it is recommended that you edit and modify the LOCKSIZE values within **EDISDB2** from ANY to ROW as indicated below. LOCKSIZE is specified on the CREATE TABLESPACE command. If you change the LOCKSIZE to ROW, you may also need to increase the maximum number of locks allowed for the tablespace.

  Heavily updated tables might include

  – Transaction store tables (EDITSxx)
  – Manangement reporting tables (EDIMRxx)
  – Event logging table (EDIELOG)
  – Trading partner profile control numbers table (EDIPROF)

- If necessary, edit and modify the storage group name and the database name to comply with your site naming conventions.

- If necessary, change definitions in the CREATE PROCEDURE statements to match your WLM Environment and Collection ID.

__ 3. Run, via SPUFI, **EDISDB2** and verify that all return codes are zero. If you receive a nonzero return code, see the appropriate product's message manual for further information. If necessary, use the scroll forward key to page through the return codes.

___ 4. Configure changes required to run WDI Stored Procedures in a DB2 WLM Application Environment. This requires changes to DB2's PROC that starts the WLM defined in the DDL.

    ___ a. Add the WDI installation load library (EDI.V3R3M0.SEDILMD1) and the XML4C load library from the XML Toolkit (SIXMMOD1) to the STEPLIB.

    ___ b. Add a DD and a dataset for EDITSIN. Allocate the dataset as fixed-block, record length 80. This supplies WDI with attachment information as described in the *WDI Programmer's Reference*. A typical EDITSIN includes text such as:

```
PLAN(EDIENU33) SYSTEM(DSN) CAF(8)
```

This specifies the DB2 plan used by WDI, the DB2 subsystem name and a parameter telling WDI to attach to DB2 using RRSAF. The CAF value is required whenever using a WLM address space.

___ 5. Add the OMVS external link for WDI's command script interpreter, EDISERVR. DB2 will execute WDI as a RACF user ID defined in the RACF started procedures table. This ID must be defined with an OMVS segment. To allow the OMVS services to access the WDI load library, place an external link in the home directory for the WLM owner ID. The syntax for the link command is similar to:

```
ln -e EDISERVR /home/wdiuser/EDISERVR
```

The command may be issued from any shell interpreter available on the z/OS host including the "omvs" shell that runs under ISPF. Case is important.

___ 6. Make sure all steps complete successfully and go to 2.1.12, "Step 12. Loading Repository DB2 Tables."

2.1.12 Step 12. Loading Repository DB2 Tables

EDIJRDLD runs the IBM DB2 utility procedure, DSNUPROC, to load default data for certain tables that are required in order for WebSphere Data Interchange to run.   The remaining default data will be loaded later using the WDI server IMPORT command. The distribution tape provides the default data. The SMP/E installation process loads the data into the target library EDI.V3R3M0.SEDIDB21.

The following steps describe how to load repository DB2 tables.

___ 1. Edit the EDIJRDLD member in EDI.V3R3M0.SEDIINS1.

Modify the member for your environment as shown in the comment at the beginning of the JCL.

___ 2. Run the job.

___ 3. Verify that all return codes are zero. If you receive a nonzero return code, see the appropriate product's message manual for further information.

___ 4. If all return codes are zero, go to 2.1.13, "Step 13. Running RUNSTATS on the Loaded DB2 Tables."

2.1.13 Step 13. Running RUNSTATS on the Loaded DB2 Tables

EDIJRNST uses the RUNSTATS command to run the DB2 DSNUPROC procedure.

EDIJRNST gathers information about the previously loaded WebSphere Data Interchange tables. The DB2 catalog contains the information.

The SQL optimizer uses the DB2 catalog to select access paths to the data during the BIND process.

The following steps describe how to run RUNSTATS on the loaded DB2 tables.

___ 1. Edit the EDIJRNST member in EDI.V3R3M0.SEDIINS1.

Modify the member for your environment as shown in the comment at the beginning of the JCL.

___ 2. Run the job.

___ 3. Verify that the return code is not greater than 0004. If you receive a return code greater than 0004, see the appropriate product's message manual for further information.

___ 4. If the return code is not greater than 004, go to 2.1.14, "Step 14. Binding the DB2 Plan."

2.1.14 Step 14. Binding the DB2 Plan

This step uses the IKJEFT01 utility to bind the WebSphere Data Interchange EDIENU33 DB2 application plan for use in TSO and batch.

The following steps describe how to bind the DB2 plan:

___ 1. Edit the EDIJBIND member in EDI.V3R3M0.SEDIINS1.   Modify the member for your environment as shown in the comment at the beginning of the JCL.

___ 2. Run the job.

___ 3. Verify that the return code is zero.   If you receive a return code greater than zero, see the appropriate product's message manual for further information.

___ 4. If the return code is zero, go to 2.1.15, "Step 15. Granting Authorization on the DB2 Objects."

2.1.15 Step 15. Granting Authorization on the DB2 Objects

After binding the WebSphere Data Interchange DB2 application plan, you must make it available to those using WebSphere Data Interchange by granting authorization on DB2 objects.

Note: SQL member EDISGRNT of the PDS EDI.V3R3M0.SEDISQL1 provides sample SQL statements for granting authorization on the DB2 objects.

The following steps describe how to grant authorization on the DB2 objects. At a minimum, you **must** authorize GRANT EXECUTE authority to WebSphere Data Interchange users on the plan.

___ 1. You can use SPUFI to run an SQL statement similar to the following example that makes plan EDIENU33 available to user ID USER01:

```
                GRANT EXECUTE ON PLAN EDIENU33 TO USER01;
```

Optionally, you can use SPUFI to run SQL statements, similar to the following example, that make views referenced by the dynamic SQL execution available to user ID USER01:

```
GRANT SELECT ON TABLE EDIENU33.EDIVMRNA1,
                      EDIENU33.EDIVMRTA1,
                      .
                      .
                      .
                      EDIENU33.EDIVMRTI3   TO USER01;
```

__ 2. If you want to give *all* DB2 users access to the plan EDIENU33 and views referenced by the dynamic SQL execution plan, enter the following SQL statement:

```
GRANT EXECUTE ON PLAN EDIENU33 TO PUBLIC;
GRANT SELECT ON TABLE EDIENU33.EDIVMRNA1,
                      EDIENU33.EDIVMRTA1
                      .
                      .
                      .
                      EDIENU33.EDIVMRTI3   TO PUBLIC;
```

Note: Dynamic SQL is used only when DYNSQL(Y) is specified during WebSphere Data Interchange utility execution for Management reporting data extracts.

__ 3. WDI Client users require SELECT, INSERT, DELETE, UPDATE authority to all tables, or you might want to grant table privileges for DB2 utility (QMF*, SPUFI, or DBEDIT) usage on the WebSphere Data Interchange tables similar to the following example:

```
GRANT SELECT, INSERT, DELETE, UPDATE ON TABLE EDIENU33.EDIADFCSTDET,
                                        EDIENU33.EDIADFDICT,
                                        .
                                        .
                                        .
                                        EDIENU33.EDIVTSIS TO USER01;
```

__ 4. Grant authorizations that are required for remote job submission:

There are at least three privileges required to execute the WebSphere Data Interchange stored procedures.

   a. First, the user connecting to the database using the WDI Client must have authorization to execute the stored procedures.   This is accomplised with a DDL statement to be executed via SPUFI similar to the following:

```
   GRANT EXECUTE ON PROCEDURE EDIADDREQ TO USER02;
   GRANT EXECUTE ON PROCEDURE EDIEXEC1  TO USER02;
   GRANT EXECUTE ON PROCEDURE EDIEXEC2  TO USER02;
```

   b. Second, the functional user ID that executes the stored procedure within DB2 must have authorization to the package collection created during the BIND PACKAGE in Step 14, Binding

the DB2 Plan. If the functional user does not have authorization, grant it with DDL similar to the following via SPUFI:

```
GRANT EXECUTE ON PACKAGE EDI33001.* TO USER02
```

c. Finally, the functional user must have the ability to execute the plan containing the WDI SQL. WDI executes using the credentials of the DB2 stored procedure region, not the credentials of the user connecting to the database. Ensure the proper authorization with DDL similar to the following via SPUFI:

```
GRANT EXECUTE ON PLAN EDIENU33 TO USER02
```

__   5. If all return codes are zero, go to 2.1.16, "Step 16. Importing Default Data Into Version 3.3."

2.1.16 Step 16. Importing Default Data Into Version 3.3

This step will import default data into the WebSphere Data Interchange Version 3.3 database.

The following steps describe how to import the default data.

__   1. Edit the EDIJDAT1 member in EDI.V3R3M0.SEDIINS1.

Modify the member for your environment as shown in the comment at the beginning of the JCL. To load the default data, always substitute "MEMBER=EDIDATA" in the JCL.

__   2. Run the job.

__   3. Verify that all return codes are zero. If you receive a nonzero return code, see messages in the JCL Job's PRTFILE.

__   4. If all return codes are zero and you are migrating from Version 3.1 (DB2) or from Version 3.2, go to 2.1.17, "Step 17a. Build Control File for Exporting Version 3.1 or Version 3.2 DB2 Data."

__   5. If all return codes are zero and you are migrating from Version 3.1 (VSAM), go to 2.1.18, "Step 17b. Build Control File for Exporting Version 3.1 VSAM Data."

__   6. If all return codes are zero and you are installing for the new customer, go to 2.1.21, "Step 19. Verifying the Installation."

2.1.17 Step 17a. Build Control File for Exporting Version 3.1 or Version 3.2 DB2 Data

EDIJXPAD executes the WebSphere Data Interchange program EDIXPEA to build the control file for exporting data. The following steps describe how to build the control file for exporting data for existing DB2 customers. A control file entry will be made for each of the following data type members:

- Standards
- ADFs
- Maps, with associated Usages and Control strings

- Profiles
- Tables

__ 1. Edit the EDIJXPAD member in EDI.V3R3M0.SEDIINS1.

   Modify the member for your environment as shown in the comment at the beginning of the JCL.

__ 2. Run the job.

__ 3. Verify that all return codes are zero. If you receive a nonzero return code, see the appropriate product's message manual for further information.

__ 4. If all return codes are zero, go to 2.1.19, "Step 18a. Migrate Version 3.1 or Version 3.2 DB2 Data to Version 3.3"

2.1.18 Step 17b. Build Control File for Exporting Version 3.1 VSAM Data

EDIJXPAV executes the WebSphere Data Interchange program EDIXPEA to build the control file for exporting data. The following steps describe how to build the control file for exporting data for existing VSAM customers. A control file entry will be made for each of the following data type members:

- Standards
- ADFs
- Maps, with associated Usages and Control strings
- Profiles
- Tables

__ 1. Edit the EDIJXPAV member in EDI.V3R3M0.SEDIINS1.

   Modify the member for your environment as shown in the comment at the beginning of the JCL.

__ 2. Run the job.

__ 3. Verify that all return codes are zero. If you receive a nonzero return code, see the appropriate product's message manual for further information.

__ 4. If all return codes are zero go to 2.1.20, "Step 18b. Migrate Version 3.1 non-DB2 Data to Version 3.3 DB2 Environment"

2.1.19 Step 18a. Migrate Version 3.1 or Version 3.2 DB2 Data to Version 3.3

This step is for Version 3.1 or Version 3.2 customers. EDIJEXPD will read the control file created previously, and migrate each of the existing data type members identified:

- Standards
- ADFs
- Maps, with associated Usages and Control strings
- Profiles
- Tables

The following steps describe how to migrate Version 3.1 or Version 3.2 customer data records using EDIJEXPD. The job EDIJRNST will need to be re-run after you run the job EDIJEXPD.

___ 1. Edit the EDIJEXPD member in EDI.V3R3M0.SEDIINS1.

   Modify the member for your environment as shown in the comment at the beginning of the JCL.

___ 2. Run the job.

___ 3. Verify that all return codes are zero. If you receive a nonzero return code, see the appropriate product's message manual for further information.

___ 4. Run the job EDIJRNST. This job was previously set-up and run in 2.1.13, "Step 13. Running RUNSTATS on the Loaded DB2 Tables." Run this job again.

___ 5. If all return codes are not greater than 0004, go to 2.1.21, "Step 19. Verifying the Installation."

2.1.20 Step 18b. Migrate Version 3.1 non-DB2 Data to Version 3.3 DB2 Environment

This step is for Version 3.1 (non-DB2) customers who are switching to DB2 while migrating to a Version 3.3 DB2 environment. EDIJEXVD will read the control file created previously, and migrate each of the existing data type members identified:

- Standards
- ADFs
- Maps, with associated Usages and Control strings
- Profiles
- Tables

The following steps describe how to migrate Version 3.1 customer non-DB2 data records to DB2 using EDIJEXVD. The job EDIJRNST will need to be re-run after you run the job EDIJEXPD.

___ 1. Edit the EDIJEXVD member in EDI.V3R3M0.SEDIINS1.

   Modify the member for your environment as shown in the comment at the beginning of the JCL.

___ 2. Run the job.

___ 3. Verify that all return codes are zero. If you receive a nonzero return code, see the appropriate product's message manual for further information.

___ 4. Run the job EDIJRNST. This job was previously set-up and run in 2.1.13, "Step 13. Running RUNSTATS on the Loaded DB2 Tables." Run this job again.

___ 5. If all return codes are not greater than 0004, go to 2.1.21, "Step 19. Verifying the Installation."

2.1.21 Step 19. Verifying the Installation

The following steps describe how to verify the installation.

1. Install WebSphere Data Interchange Client. WebShpere Data Interchange Client is a Microsoft„ Windows-based interface for servers. This interface permits you to use a PC to create and maintain WebSphere Data Interchange profiles, EDI standards, data formats, and maps. It also can be used to load XML DTDs into WebSphere Data Interchange.

   Note that WebSphere Data Interchange Client does not translate data. All translation functions are performed by the WebSphere Data Interchange Server. WebSphere Data Interchange Client provides the ease of use inherent in Windows„ to make your setup, mapping, and administration faster and easier.

   Servers are supported on the following operating systems:

   - AIX(R), Version 5.2
   - Windows
   - z/OS Version 1.6 (ESA mode), or later (5694-A01)

2. Setup Client Access to the z/OS database client/server mode.

   For More information on WebSphere Data Interchange Client installation see the WebSphere Data Interchange User's Guide.

3. The z/OS default load data contains Data Transformation maps, functional acknowledgment maps, service segment validation maps, and the related objects (EDI Standards, XML DTD, Trading Partner Profile, sample network adapters, and environment profiles).

   Installation may be verified by viewing or opening these objects. Do not change or save any changes to the default load data objects.

4. If you are migrating from a previous version, view or open each object type that would have been migrated from the previous version.

If you are a WebSphere Data Interchange for CICS customer, review the operational changes discussed in 2.1.22, "Step 20. Operational Changes," and then go to Chapter 3, "Installing WebSphere Data Interchange for CICS Transaction Server for z/OS."

This step completes the WebSphere Data Interchange for z/OS installation. However, if you are migrating from a previous release, review the operational changes discussed in 2.1.22, "Step 20. Operational Changes."

2.1.22 Step 20. Operational Changes

If you are migrating from a previous version of WebSphere Data Interchange, note the following operational changes.

The Administrator's Menu has been removed from the z/OS interface. These functions are now performed using WebSphere Data Interchange Client Version 3, Release 3. Export and import functions are available in batch or from a CICS task.

WDI Common Event Handler Java API Support. The common events handler enables you to improve the performance of the WebSphere Data Interchange Event Log and Print File and enables you to develop custom Java plug-in programs to process WebSphere Data Interchange generated Events. See the *WDI Programmer's Reference* for more information. Additional install steps are required for this function. See 2.1.24, "Step 22. WDI Common Event Handler Java API Support (Optional)" on page 38.

2.1.22.1 Customers migrating from Version 3, Release 1:

Network Operation profiles (NETOP) from DataInterchange V3.1 and earlier are not compatible with Version 3 Release 3. They will not migrate during this install and should never be imported. Version 3 Release 3 installs the Network Operation profiles required to support all Network profiles (NETPROF) shipped with the DataInterchange product.

Operational objects such as Maps, EDI Standards and Application Data Formats that existed under the z/OS interface were converted and imported during this installation of Version 3.3. These objects can only be accessed via the WebSphere Data Interchange Client Version 3 Release 3. (For best results, use DB2 Connect with the Client interface to access the common database.) As a result of this conversion, particular attention should be paid to the relationship between a map and its EDI standard. When a map was created via the z/OS interface back on Version 3, Release 1, the EDI standard object at that time was copied into its definition. Hence, subsequent changes to the standard object were not reflected immediately within the map. On Version 3 Release 3 this is no longer the case. An existing map is dependent on the current standard definition. Any change to the standard are immediately effective in a WebSphere Data Interchange Client map. However, the control string remains unchanged until recompilation. Thus, mapping migration is no longer a function, but a natural occurrence when the user changes a map's reference to a given standard For the latest information regarding conversion of Maps, EDI Standards and Application Data Formats, please refer to member EDI31DOC in SMP/E target library EDI.V3R3M0.SEDIINS1.

Any-to-any translation is a new WebSphere Data Interchange feature that allows translation from any supported source document type to any supported target document type. Supported document types include data formats, EDI standards, and XML data. This vastly expands the translation capabilities to a greater number of formats. Any-to-any translation uses a new type of mapping called a data transformation map. This is a set of instructions that describes how to translate data from a source document into a target document. Both the source and target documents can be any one of several supported document types (application data to EDI standard format, EDI standard format to application data, XML data to EDI standard format, and so on). Data transformation maps use the TRANSFORM

command to do any-to-any translation. You should use any-to-any translation with new maps that you create. This is now the recommended method for translating documents. For more information about the TRANSFORM command, refer to the WebSphere Data Interchange Programmer's Reference. Data transformation maps are created using WebSphere Data Interchange Client. For more information about the mapping features in the Client, refer to the WebSphere Data Interchange for Multiplatforms Mapping Guide. PERFORM commands from prior releases are still supported on this release.

JCL changes between Version 3.1 and Version 3.3:

- A sample utility job, EDIUTILD, is included in library EDI.V3R2M0.SEDISAM1.
  Customers migrating from a VSAM database should use this new member to
  recreate JCL.

- The PLAN keyword in the SYSTSIN DD (or the EDITSIN DD) has changed from EDIENU31
  to EDIENU33.

- All data set names with V3R1M0 have been changed to V3R3M0.

- Transaction Store, Management Reporting, and SAP status migration
  is not available.

Several WebSphere Data Interchange control blocks were changed for Version 3.3. These changes will require, at a minimum, that all user applications using these WebSphere Data Interchange control blocks be recompiled. Due to a code change within the translator on DataInterchange Version 3.1, made specifically to the Translate file API (ZFCBFUNC=212), it is now necessary to set TRCB-FASPEC to "Y" in the event a translator API program is setting TRCB-FUNACKFLE to a non-blank value. For other control block changes, see Table 9 for a list of these other API control block changes.

2.1.22.2 Field Size Changes between V3.1 and V3.3

Table 9. COBOL Field Size Changes between V3.1 and V3.3

Block	Old Field Size	New Field Size
CMNP-DATA	CMTP-LOGLOCK PIC X(1).	CMTP-LOGLOCK PIC X(17).
CMRQ-DATA	CMRQ-LOGLOCK PIC X(1).	CMRQ-LOGLOCK PIC X(17).
CMSP-DATA	CMTP-LOGLOCK PIC X(1).	CMTP-LOGLOCK PIC X(17).
CMTP-DATA	CMTP-LOGLOCK PIC X(1).	CMTP-LOGLOCK PIC X(17).

Note: The PLI, C, and Assembler blocks have the same field size changes.

2.1.22.3 Customers migrating from Version 3.2

Transaction Store, Management Reporting, and SAP status migration is available.   See APPENDIX B for more information.

Data Transformation Functional Acknowledgment maps and Service Segment Validation maps that are shipped with WDI have been updated for Version 3.3. The map names and associated object names have also changed. WDI shipped objects beginning with the '$' character will begin with the '&' character. During migration, Data Transformation Rules specifying the WDI shipped maps will be changed to use the maps beginning with the '&' character. If you have customized the shipped maps and want to use them with WDI V3.3, then you will need to either customize the shipped maps again or remove the shipped maps and replace them with your customized versions.

JCL changes between V3.2 or V3.2.1 and V3.3:

- The PLAN keyword in the SYSTSIN DD (or the EDITSIN DD) has changed from EDIENU32 to EDIENU33.

- All data set names with V3R2M0 have been changed to V3R3M0.

2.1.23 Step 21. Migrating Operational Data

If you are migrating from Version 3.2, you can also migrate your Transaction Store, Management Reporting, and SAP status data from your Version 3.2 system to your Version 3.3 system. You can do this migration as a part of your initial installation, or you can wait until you are ready to start using your Version 3.3 system for production data.   Migration of Transaction Store, Management Reporting, and SAP status data from Version 3.1 is **not** supported.

For more information on the steps required to migrate this operational data, see Appendix B on page 63.

2.1.24 Step 22. WDI Common Event Handler Java API Support (Optional)

Ensure java is installed on the Host. IBM SDK for z/OS, Java 2 Technology Edition, Version 1.4 SR(6) or higher is required.   To verify, type on the ISPF Command screen:

```
omvs
java -version
```

Verify the version displayed is at least:

```
Java(TM) 2 Runtime Environment, Standard Edition (build 1.4.2)
Classic VM (build 1.4.2, J2RE 1.4.2 IBM z/OS Persistent Reusable VM
build cm142-20060824 (SR6) (JIT enabled: jitc))
```

Then type:

```
exit
```

1. Unix System Services will be required for this step. To start Unix System Services, from z/OS ISPF, select the option to enter TSO or workstation commands. Enter command OMVS. Set up a HFS home directory that will contain the WebSphere Data Interchange java jars and property files. Use the **mkdir** command in Unix System Services to create the /u/edi/java HFS home directory, then exit UNIX System Services.   Beginning on the ISPF Command screen, enter:

```
omvs
mkdir /u/edi/java
exit
```

2. Copy the common event handling files into the home directory. The following common event handling files are required:

   - edicevh.jar - Common Event Handling Java API
   - edicevhsample.jar - Sample email notification event handler with java source
   - env.profile - Java runtime environment variables for Java in batch JCL
   - wdi.properties - default Common Event Handling properties

   From the ISPF Command screen, enter:

```
oput 'edi.v3r3m0.sedihfs2(fxxcevhj)' '/u/edi/java/edicevh.jar'  binary
oput 'edi.v3r3m0.sedihfs2(fxxcevhs)' '/u/edi/java/edicevhsample.jar' binary
```

   The Java environment profile for the Common Event Handling Java API is the file **env.profile**. This file is used by the JVM in JCL batch z/OS. The sample batch program **EDIJCEH** runs the JVM and uses the env.profile to determine the PATH, CLASSPATH and JAVA_HOME environment variables. This file will need to be customized.

```
oput 'edi.v3r3m0.sedihfs2(fxxprof)' '/u/edi/java/env.profile' text
```

   The properties file for the java common event handling plugins is wdi.properties

```
oput 'edi.v3r3m0.sedihfs2(fxxprop)' '/u/edi/java/wdi.properties' text
```

3. Customization for JVM properties file.   You will need to update the CLASSPATH value in the env.profile file to point to the Common Event Handling API jars.

   For the sample email notification you will need the following files:

```
activation.jar
mail.jar
```

   These jars are part of the Sun JavaMail API and Sun JavaBeans Activation Framework downloadable from **http://www.sun.com**. The CLASSPATH entries for these need to be added.

   The CLASSPATH must have the IBM Java for z/OS recjava.jar and recordio.jar locations. These are found in the lib/ext subdirectory of IBM Java for z/OS. For example if IBM Java for z/OS path is:

```
/java/IBM/J1.4
```

then the CLASSPATH entries would be

```
/java/IBM/J1.4/lib/ext/recordio.jar:/java/IBM/J1.4/lib/ext/recjava.jar:
```

The CLASSPATH values in env.profile will need to be changed according to your particular installation, for example:

```
CLASSPATH=/u/edi/java:
/u/edi/java/activation.jar:
/usr/lpp/java/J1.4/lib/ext/recjava.jar:
/usr/lpp/java/J1.4/lib/ext/recordio.jar:
/u/edi/java/mail.jar:
/u/edi/java/edicevh.jar:
/u/edi/java/edicevhsample.jar:
/u/edi/java/wdi.properties:.
JAVA_HOME=/usr/lpp/java/J1.4

PATH=$PATH:/usr/lpp/java/J1.4/bin
```

PATH must have the path to the location of the IBM Java for z/OS runtime. Your particular PATH may be different than the sample provided. Update this value according to your particular installation.

JAVA_HOME is the location of the root IBM Java for the z/OS install directory where the Java runtime bin and lib directories are located. Your particular JAVA HOME path may be different than the sample provided. Update this value accordingly.

To open files for editing, from z/OS ISPF, select option to enter TSO or workstation commands and use the command OEDIT. For example: oedit /u/edi/java/env.profile

Note: The characters ":." (colon/period) must end the CLASSPATH identification before the JAVA_HOME subdirectory.

Note: CLASSPATH entries provided above are broken into separate lines for ease of reading. The real env.profile requires the entire CLASSPATH entry as one record whereby each entry is concatenated together as is demonstrate in the sample env.profile provided.

Hint: If you find you need more than 255 bytes when 'OEDITING' the env.profile in OMVS or for ease of editing, one suggestion is to ftp the file env.profile from OMVS to a workstation in ASCII. Open up a Windows editor of your choice (ie., Notepad) and turn 'on' Word Wrap. This will 'wrap' the one long CLASSPATH line into several lines all in your 'view'. You can then edit the path for your environment and not have to shift from left to right. When finished with your updates, save the file on the workstation and ftp the file env.profile back to the location in OMVS on your Host system.

4. Verify the installation by executing the sample JCL provided with WebSphere Data Interchange by running the member EDIJCEH in EDI.V3R3M0.SEDISAM1. You must first change the EDIJCEH JCL job cards for your particular environment. The verification sample Java API reads the PRTFILE and runs a simple event handler. Read the Job execution log. The message:

```
*** WDI Event Handler Installation Verified ***
```

verifies that the Java API was correctly installed, configured, and executed.

After verifying the installation in this step, the sample email notification program will be used to do step 5.

5. Customization for the WDI email notification. Update the wdi.properties file. The SMPT email server URL or IP address, and the email account id information must be known in advance.

   Open the wdi.properties file for editing. To open files for editing, from z/OS ISPF, select option to enter TSO or workstation commands and use the command **OEDIT**. Locate and set the following property values:

   - The Event Handler Plugin class - Comment out the DummyEventHandler wdiHandler setting and remove the comment for WDIEmailNotify. (The '#' character is used for commenting out a line, remove to uncomment)

     ```
     wdiHandler=com.ibm.edi.wdievents.email.WDIEmailNotify
     #wdiHandler=com.ibm.edi.wdievents.email.DummyEventHandler
     ```

   - The URL or IP address of your smtp server - This should be the URL or IP address of your SMTP (Simple Mail Transfer Protocol) email server. smtpServerAddress is the address of the SMTP server for outgoing messages. You get this information from your LAN administrator or internet service provider (ISP):

     ```
     smtpServerAddress=nnn.nnn.nnn.nnn
     ```

     The FROM address is the address of the email sent. The TO address is the address to receive the notification.

   - The from address of the email notification sent -

     ```
     fromAddress=host@myaddress.com
     ```

   - The destination email address to receive the notification -

     ```
     toAddresses=admin@address.com
     ```

   If your SMTP server requires user ID and password authentication, you may specify these in the userId and password properties. Uncomment these and change them if necessary. Otherwise, leave them commented out. These values can be entered here or in the Event Destination Profile.

   ```
   userId=my_id
   password=my_password
   ```

   Verify the customization by executing the sample JCL provided with WebSphere Data Interchange by running the member EDIJCEH in EDI.V3R3M0.SEDISAM1.

   Problem Determination:

   If you are not getting email. check the /u/edi/java/wdi.properties file, the /u/edi/java/env.profile file, and the sample JCL EDIJCEH to ensure file paths have been updated with the directory that was used during the install steps are correct. Check the install directory to be sure all required *.jar files are available.

   Logging and tracing information defaults to output STDOUT and STRERR. In the wdi.properties file, the logging and tracing output for the email plug-in can be redirected.

   ```
   #Logging and Tracing
   logFile=wdiemail.log
   saveLogDirectory=./
   debug=false
   ```

The /u/edi/java/wdi.properties file contains a debug statement. This is set to false. By setting this flag to true you can trace information as a problem determination aid for connection problems with the server. If the problems persist, contact your WDI Support Center.

A few basic OMVS or USS commands for customers not familiar with Unix System Service are listed below:

- **cd** change directory. Example: cd /u/edi/java
- **chmod** change the permissions of directories or files. Example: chmod +rwx wdi.properties (+rwx=read/write/execute). (alternately, use chmod 755 wdi.properties)
- **ls** list files,
- **ls -l** list files with attributes - date, time, permissions
- **obrowse** view file without editing Example: obrowse wdi.properties
- **oedit** edit a file Example: oedit wdi.properties
- **more** view Contents of a file. Example: more wdi.properties
- **rm** remove a directory or file Example: rm sample.file
- **rm -R** remove a directory or file recursively Example: rm -R sample (directory)
- **mkdir** creates a directory Example: mkdir sample.file
- **quit** Quit more command
- **df** display filesystems mounted to USS.
- **mv** move or rename file Example: mv wdi.properties wdi2.properties
- **cp** copy a file Example: cp profile.sh profile.sh.old
- **exit** Exit USS (alternately can can press F2 and type quit)

Additional information about all the properties and their values is in the *WebSphere Data Interchange V3.3 Programmer's Reference* and *WebSphere V3.3 User Guide*.

Chapter 3. Installing WebSphere Data Interchange for CICS Transaction Server for z/OS

Chapter 3. Installing WebSphere Data Interchange for CICS Transaction Server for z/OS

This chapter contains the steps you follow to complete the installation of WebSphere Data Interchange for CICS Transaction Server for z/OS and to verify that installation was successful. Prior to continuing, you must have completed Steps 1 through 21. For more information, see Chapter 2, "Installing WebSphere Data Interchange."

3.1.1 Special Considerations for LE and C++ With CICS

C/C++ is required for WDI to run in the CICS environment.

Refer to section "Installing CICS Suport for Language Environment" in the *CICS Transaction Server for z/OS Installation Guide* for information about enabling LE in a CICS environment.

Refer to section "Language Environment support for C and C++" in the *CICS Transaction Server for z/OS Installation Guide* for information about enabling C++ in a CICS environment.

Refer to section "Preparing CICS for Use with z/OS Language Environment" in the *z/OS C/C++ Programming Guide* for information about CSD, DFHRPL, and I/O Stream Library considerations. The library containing iostream (SCLBDLL or its equivalent) must be added to the DFHRPL concatenation.

3.2 Installation Steps

Each step begins with a description of the action.

Notes:

1. You must complete the installation for WebSphere Data Interchange for z/OS Version 3.3 before installing the WebSphere Data Interchange for CICS portion.

2. The distribution tape provides a set of CICS sample tables and JCL. The SMP/E installation process loads the tables and JCL into the target library, EDI.V3R3M0.SEDISAM2.

3.2.1 Step 23. Creating COBOL User Exit Enabling Module for CICS

FXXJLCBL creates COBOL user exit enabling module for customers who write user exits in COBOL. FXXJLCB2 creates COBOL II user exit enabling module for customers who write user exits in COBOL II.

Note: If you do not write user exits in COBOL or COBOL II, go to 3.2.2, "Step 24. Creating the EDIW Transaction Module for CICS."

The following steps describe how to create the module.

__ 1. Edit as appropriate, either the FXXJLCBL or FXXJLCB2 member in EDI.V3R3M0.SEDIINS2.

Modify the member for your environment as shown in the comment at the beginning of the JCL.

___ 2. Run the job.

___ 3. Verify that all return codes are not greater than 0004. If you receive a return code greater than 0004, see the appropriate product's message manual for further information.

___ 4. If all return codes are not greater than 0004, go to 3.2.2, "Step 24. Creating the EDIW Transaction Module for CICS."

3.2.2 Step 24. Creating the EDIW Transaction Module for CICS

FXXJEDIW creates the EDIW transaction module for CICS.

The following steps describe how to create the module.

___ 1. Edit the FXXJEDIW member in EDI.V3R3M0.SEDIINS2.

Modify the member for your environment as shown in the comment at the beginning of the JCL.

___ 2. Run the job.

___ 3. Verify that all return codes are not greater than 0004. If you receive a return code greater than 0004, see the appropriate product's message manual for further information.

___ 4. If all return codes are not greater than 0004, go to 3.2.3, "Step 25. Creating DB2 Synonyms for CICS."

3.2.3 Step 25. Creating DB2 Synonyms for CICS

FXXSSYNO uses the DB2 Structured Query Language (SQL) Processor Using File Input (SPUFI) facility under Interactive System Productivity Facility/Program Development Facility (ISPF/PDF) to create the DB2 synonyms for CICS. Members in the distributed data set, EDI.V3R3M0.SEDISQL2, contain the SQL statements.

Note: Do not do this step if the user ID that created the DB2 objects is the same as the user ID binding the DB2 plan for CICS.

The following steps describe how to access SQL statements and submit them to DB2 for processing, using the DB2 SPUFI option under ISPF/PDF.

___ 1. On the DB2I Primary Option Menu, select the SPUFI option (normally option 1).

___ 2. On the SPUFI panel, enter

```
EDI.V3R3M0.SEDISQL2(FXXSSYNO)
```

This updates the SQL statements on the panel. The SPUFI panel contains comments that indicate any required changes.

___ 3. If necessary, change the table names so they match those that are created during WebSphere Data Interchange/MVS installation.

___ 4. Run the job.

___ 5. Verify that all return codes are zero. If you receive a nonzero return code, see the appropriate product's message manual for further information. If necessary, use the scroll forward key to page through the return codes.

___ 6. If all return codes are zero, go to 3.2.4, "Step 26. Binding the DB2 Plan."

3.2.4 Step 26. Binding the DB2 Plan

This step uses the IKJEFT01 utility to bind the WebSphere Data Interchange DIENU32C DB2 application plan for use in TSO and batch.

The following list describes the steps to bind the DB2 plan.

___ 1. Edit the FXXJBIND member in EDI.V3R2M0.SEDIINS2. Modify the member for your environment as shown in the comment at the beginning of the JCL.

___ 2. Run the job.

___ 3. Verify that the return code is zero.   If you receive a return code greater than zero, see the appropriate product's message manual for further information.

___ 4. If the return code is zero, go to 3.2.5, "Step 27. Granting Authorization on the DB2 Objects."

3.2.5 Step 27. Granting Authorization on the DB2 Objects

After binding the WebSphere Data Interchange DB2 application plan, you must make it available to those who will be using WebSphere Data Interchange by granting authorization.

The following list describes the commands used when granting authorization on the DB2 objects used in a CICS region. The command that you use depends on the results you desire.

___ 1. You can use SPUFI to run an SQL statement similar to the following example that makes plan DIENU33C available to user ID USER01:

```
GRANT EXECUTE ON PLAN DIENU33C TO USER01;
```

___ 2. If you want to give all DB2 users access to the DIENU33C plan, enter the following SQL statement:

```
GRANT EXECUTE ON PLAN DIENU33C TO PUBLIC;
```

___ 3. Go to 3.2.6, "Step 28. Setting Up the DCT for WebSphere Data Interchange."

3.2.6 Step 28. Setting Up the DCT for WebSphere Data Interchange

FXXODCT contains WebSphere Data Interchange's transient data queue defintions. Complete the following steps to add these defintions to your CICS System Definition (CSD) file. The default Resource Definition Online (RDO) group is EDIGROUP. You may change this as appropriate. If necessary, see the *CICS Resource Definition Guide* for information regarding the CSD file utility program, DFHCSDUP, and transient data queue defintion.

The following steps describe how to add Websphere Data Interchange's transient data queues to your CICS System Defintion (CSD) file:

__ 1. Edit the FXXODCT member in EDI.V3R3M0.SEDISAM2.

   Modify the member for your environment as described in the comments at the beginning of the JCL.

__ 2. Run the job.

__ 3. Verify that all return codes are zero. If you receive a nonzero return code, see the appropriate product's message manual for further information.

__ 4. If all return codes are zero, go to 3.2.7, "Step 29. Setting Up the PCT for WebSphere Data Interchange."

3.2.7 Step 29. Setting Up the PCT for WebSphere Data Interchange

FXXOPCT contains WebSphere Data Interchange's CICS transaction defintions. Complete the following steps to add these defintions to your CICS System Definition (CSD) file. The default Resource Definition Online (RDO) group is EDIGROUP. You may change this as appropriate. If necessary, see the *CICS Resource Definition Guide* for information regarding the CSD file utility program, DFHCSDUP, and transaction defintion.

The following steps describe how to add Websphere Data Interchange's CICS transactions to your CICS System Defintion (CSD) file:

__ 1. Edit the FXXOPCT member in EDI.V3R3M0.SEDISAM2.

   Modify the member for your environment as shown in the comments at the beginning of the JCL.

__ 2. Run the job.

__ 3. Verify that all return codes are zero. If you receive a nonzero return code, see the appropriate product's message manual for further information.

__ 4. If all return codes are zero, go to 3.2.8, "Step 30. Setting Up the PPT for WebSphere Data Interchange."

3.2.8 Step 30. Setting Up the PPT for WebSphere Data Interchange

FXXOPPT contains WebSphere Data Interchange's CICS program defintions. Complete the following steps to add these defintions to your CICS System Definition (CSD) file. The default Resource Definition Online (RDO) group is EDIGROUP. You may change this as appropriate. If necessary, see the *CICS Resource Definition Guide* for information regarding the CSD file utility program, DFHCSDUP, and program defintion.

The following steps describe how to add Websphere Data Interchange's CICS programs to your CICS System Defintion (CSD) file:

__ 1. Edit the FXXOPPT member in EDI.V3R3M0.SEDISAM2.

Modify the member for your environment as shown in the comments at the beginning of the JCL.

__ 2. Run the job.

__ 3. Verify that all return codes are zero. If you receive a nonzero return code, see the appropriate product's message manual for further information.

__ 4. If all return codes are zero, go to 3.2.9, "Step 31. Setting Up the RCT for WebSphere Data Interchange."

3.2.9 Step 31. Setting Up the RCT for WebSphere Data Interchange

FXXORCT contains WebSphere Data Interchange's CICS DB2 defintions. Complete the following steps to add these defintions to your CICS System Definition (CSD) file. The default Resource Definition Online (RDO) group is EDIGROUP. You may change this as appropriate. If necessary, see the *CICS Resource Definition Guide* for information regarding the CSD file utility program, DFHCSDUP, and DB2CONN, DB2ENTRY, and DB2TRAN defintion.

The following steps describe how to add WebSphere Data Interchange's DB2 definitions to your CICS System Defintion (CSD) file:

__ 1. Edit the FXXORCT member in EDI.V3R3M0.SEDISAM2.

Modify the member for your environment as shown in the comments at the beginning of the table.

__ 2. Verify that all return codes are zero. If you receive a nonzero return code, see the appropriate product's message manual for further information.

__ 3. If all return codes are zero, go to 3.2.10, "Step 32. PLT Considerations."

3.2.10 Step 32. PLT Considerations

There are several CICS Program List Table (PLT) considerations for your WebSphere Data Interchange CICS environment. None of these are required. You may have a CICS initialization PLT (SIT parm PLTPI). This table lists application programs to be executed during CICS startup. Also, you may have a CICS termination PLT (SIT parm PLTSD). This table lists application programs to be executed during CICS shutdown.

The following are WebSphere Data Interchange programs that you may choose to include in the second pass (after PROGRAM=DFHDELIM) of the CICS initialization PLT:

- If you want your WebSphere Data Interchange - GXS Expedite/CICS Continuous Receives automatically started during CICS startup, add:

    – DFHPLT TYPE=ENTRY,PROGRAM=EDICRTS

    Note: There are two post-initialization PLT programs that affect continuous receive processing. All customers should use the Expedite/CICS PLT program called EXPOSTRT that re-establishes both Information Exchange and continuous receive sessions automatically during CICS start-up, even after an immediate shutdown. The second program is an optional WebSphere Data Interchange PLT program called EDICRTS that issues the EDIR transaction to start all continuous receive profile members with an Active flag set to Y. For most customers, using EDICRTS is unnecessary because EXPOSTRT will reestablish continuous receive sessions for you. EDICRTS starts entirely new sessions, which is redundant if you continuous receive sessions were already active when the region was brought down. See the *WDI Programmer's Reference*, section Program list table considerations.

- If you do XML translations using WebSphere Data Interchange and you want to automatically start the XML parsers, add:

    – DFHPLT TYPE=ENTRY,PROGRAM=EDIJSTRT

The following are WebSphere Data Interchange programs that you may choose to include in the first pass (before PROGRAM=DFHDELIM) of the CICS termination PLT:

- If you want your WebSphere Data Interchange - GXS Expedite/CICS Continuous Receives automatically stopped during CICS shutdown, add:

    – DFHPLT TYPE=ENTRY,PROGRAM=EDICRSP

    Note: WebSphere Data Interchange also provides an optional pre-termination PLT program named EDICRSP that issues an EDIS to stop all continuous receives, and also issues a CLOSE MAILBOX request against all involved mailboxes. During a normal shutdown, EDICRSP ends all continuous receive sessions and all Information Exchange sessions. This releases all mailboxes using continuous receive so that other communication applications requiring the same mailboxes can execute while the CICS region is down. This will prevent the Information Exchange session problems discussed earlier. However, if you have no other Information Exchange communication applications, or if the other applications you run do not use the same mailbox, you do not need to release the Information Exchange session, and PLT program EDICRSP is not needed. See the *WDI Programmer's Reference*, section "Program list table considerations."

- Program EDICRSP also executes program EDIXSOX, which quiescesWebSphere Data Interchange. If you do not specify EDICRSP and you would like to quiesce WebSphere Data Interchange during CICS shutdown, add:

    – DFHPLT TYPE=ENTRY,PROGRAM=EDIXSOX

- If you do XML translations using WebSphere Data Interchange and you want to automatically stop the XML parsers, add:

– DFHPLT TYPE=ENTRY,PROGRAM=EDIJSTOP

Go to 3.2.11, "Step 33. Setting Up the XLT for WebSphere Data Interchange and GXS Expedite/CICS."

3.2.11 Step 33. Setting Up the XLT for WebSphere Data Interchange and GXS Expedite/CICS

This step is optional. It pretains to installations that use WebSphere Data Interchange Continuous Receives with GXS Expedite/CICS and program EDICRSP is added to the CICS termination Program List Table (PLT) in order to stop those Continuous Receives when CICS terminates. If this does not pretain to your installation, go to 3.2.12, "Step 34. Setting Up XML Support for WebSphere Data Interchange."

FXXOXLT contains an alteration to WebSphere Data Interchange transaction (EDIS) and to GXS Expedite/CICS transaction (ISC2) that enables them to be run during CICS shutdown. The default Resource Definition Online (RDO) group for the WebSphere Data Interchange transaction is EDIGROUP, and the default group for the GXS Expedite/CICS transaction is EXPGROUP. You may change these as appropriate. If necessary, see the *CICS Resource Definition Guide* for information regarding the CSD file utility program, DFHCSDUP and the SHUTDOWN attribute.

The following steps describe how to enable shutdown for transactions EDIS and ISC2 in your CICS System Defintion (CSD) file:

___ 1. Edit the FXXOXLT member in EDI.V3R3M0.SEDISAM2.

Modify the member for your environment as shown in the comment at the beginning of the table.

___ 2. Verify that all return codes are zero. If you receive a nonzero return code, see the appropriate product's message manual for further information.

___ 3. If all return codes are zero, go to step 3.2.12, "Step 34. Setting Up XML Support for WebSphere Data Interchange."

3.2.12 Step 34. Setting Up XML Support for WebSphere Data Interchange

This step is required only if you are planning to use the XML support provided in WebSphere Data Interchange. WebSphere Data Interchange V3.3 uses the XML parsing APIs that are provided in Java 1.4 and later for processing XML data.

Ensure Java is installed on the host. For CICS Transaction Server 2.3, you should use JDK 1.4.

Configure the CICS/TS region for Java. See the appropriate CICS product manuals for explaination regarding this.

The following steps describe how to define and initialize the WebSphere Data Interchange XML support files:

__ 1. Create a WebSphere Data Interchange CICS home directory named /u/edi/cicshome33. You may choose a different directory name than this, but the following instructions use /u/edi/cicshome33.

Beginning on the ISPF Command screen, enter:

```
omvs
mkdir /u/edi/cicshome33
exit
```

__ 2. Copy the WebSphere Data Interchange Java parser jar file into the home directory. From the ISPF Command screen, enter:

```
oput 'edi.v3r3m0.sedihfs2(fxxsdk14)' '/u/edi/cicshome33/edijavaparser-sdk14.jar' binary
```

__ 3. Make sure JVM profile DFHJVMPR is in the WebSphere Data Interchange CICS home directory /u/edi/cicshome33. Although it is possible to use a JVM profile with another name, this setup is most straightforward if DFHJVMPR is used. It is necessary to locate DFHJVMPR in your HFS filesystem and copy it to /u/edi/cicshome33, or create a new DFHJVMPR in /u/edi/cicshome33. See the *CICS System Definiton Guide and the Java Applications In CICS* manual for information about locating or creating the JVM profile. If DFHJVMPR is located in another directory, copy it to the WebSphere Data Interchange CICS home directory. For example, if DFHJVMPR is found in /usr/lpp/cicsts/cicsts23/JVMProfiles, then beginning on the ISPF Command screen, enter:

```
omvs
cp /usr/lpp/cicsts/cicsts23/JVMProfiles/DFHJVMPR /u/edi/cicshome33
exit
```

__ 4. Change or add the WORK_DIR environment variable in JVM profile DFHJVMPR. WORK_DIR must equal /u/edi/cicshome33. From the ISPF Command screen, enter:

```
oedit
  Directory ===> /u/edi/cicshome33
  Filename ===> DFHJVMPR

WORK_DIR=/u/edi/cicshome33
```

__ 5. Add the WebSphere Data Interchange Java parser jar file to the CLASSPATH environment variable in JVM profile DFHJVMPR. Also, verify that the CICS paths are included. The actual names of the CICS paths will most likely be different in your environment. Change these appropriately. The last path name in the list should end with ":.". From the ISPF Command screen, enter:

```
oedit
  Directory ===> /u/edi/cicshome33
  Filename ===> DFHJVMPR

CLASSPATH=\
  /usr/lpp/cicsts/cicsts23/lib/dfjwrap.jar:\
  /usr/lpp/cicsts/cicsts23/lib/dfjcics.jar:\
  /u/edi/cicshome33/edijavaparser-sdk14.jar:.
```

__ 6. Add or update the CICS System Initialization Table (SIT) JVMPROFILEDIR parameter. JVMPROFILEDIR must be set to the WebSphere Data Interchange CICS home directory /u/edi/cicshome33. Set it now:

```
JVMPROFILEDIR=/u/edi/cicshome33
```

___ 7. Copy the WebSphere Data Interchange XML parser properties file. EDI.V3R3M0.SEDISAM2(FXXPARSE) is the default version of this file. The property values in this file may be changed. Whether or not this file is changed, it is necessary to copy the file to the CICS home directory as file EDIParser.properties. (EDIParser.properties must reside in the directory named in WORK_DIR.) From the ISPF Command screen enter command:

```
oput 'edi.v3r3m0.sedisam2(fxxparse)' '/u/edi/cicshome33/EDIParser.properties' text
```

Note: If you changed the DTD and trace directories in the parser properties file, then you will need to adjust the instructions below accordingly.

___ 8. Create two subdirectories in the home directory.

- Directory for dtds: /u/edi/cicshome33/dtds

- Directory for traces: /u/edi/cicshome33/traces

___ 9. Place all DTDs to be used in the dtds directory. You do not need to put anything in the traces directory. The EDI Java parsers will write any trace output to the trace directory.

___ 10. Ensure that the user ID that the CICS region will be running under has the proper permissions to access the files and directories that have been created. For the jar files and the dtds it needs read permission, and it needs rwx (read/write/execute) to all the directories (the home directory along with the dtds and traces directories).

___ 11. Go to 3.2.13, "Step 35. Setting Up Error Notification Support for WebSphere Data Interchange."

3.2.13 Step 35. Setting Up Error Notification Support for WebSphere Data Interchange

This step is required only if you are planning to use the Error Notification (Common Event Handler/E-Mail Plug-in) support provided in WebSphere Data Interchange Error notification can occur when a specific error or error severity is encountered during a WebSphere Data Interchange translation, transformation, or other Utility function.

Ensure Java is installed on the host. For CICS Transaction Server 2.3 and 3.1, you should use JDK 1.4 or higher.

Configure the CICS/TS region for Java. See the appropriate CICS product manuals for information regarding this.

The following steps describe how to define and initialize the WebSphere Data Interchange Error Notification support files:

___ 1. Create a WebSphere Data Interchange CICS home directory named /u/edi/cicshome33. This may have already been done in the "Setting Up XML Support for WebSphere Data Interchange" step. You may choose a different directory name than this, but the following instructions use /u/edi/cicshome33. Beginning on the ISPF Command screen, enter:

```
omvs
mkdir /u/edi/cicshome33
exit
```

___ 2. Copy the WebSphere Data Interchange Error Notification files into the home directory. From the ISPF Command screen, enter:

```
oput 'edi.v3r3m0.sedihfs2(fxxcevhj)' '/u/edi/cicshome33/edicevh.jar' binary
oput 'edi.v3r3m0.sedihfs2(fxxcevhs)' '/u/edi/cicshome33/edicevhsample.jar' binary
oput 'edi.v3r3m0.sedihfs2(fxxprop)' '/u/edi/cicshome33/wdi.properties text
```

___ 3. Make sure JVM profile DFHJVMPR is in the WebSphere Data Interchange CICS home directory /u/edi/cicshome33. This may have already been done in the "Setting Up XML Support for WebSphere Data Interchange" step. If so, skip this step. Although it is possible to use a JVM profile with another name, this setup is most straightforward if DFHJVMPR is used. It is necessary to locate DFHJVMPR in your HFS filesystem and copy it to /u/edi/cicshome33, or create a new DFHJVMPR in /u/edi/cicshome33. See the *CICS System Definiton Guide* and the *Java Applications In CICS* manual for information about locating or creating the JVM profile. If DFHJVMPR is located in another directory, copy it to the WebSphere Data Interchange CICS home directory. For example, if DFHJVMPR is found in /usr/lpp/cicsts/cicsts23/JVMProfiles, then beginning on the ISPF Command screen, enter:

```
omvs
cp /usr/lpp/cicsts/cicsts23/JVMProfiles/DFHJVMPR /u/edi/cicshome33
exit
```

___ 4. Change or add the WORK_DIR environment variable in JVM profile DFHJVMPR. WORK_DIR must equal /u/edi/cicshome33. From the ISPF Command screen, enter:

```
oedit
  Directory ===> /u/edi/cicshome33
  Filename ===> DFHJVMPR

WORK_DIR=/u/edi/cicshome33
```

___ 5. Add new WebSphere Data Interchange Error Notification paths to the CLASSPATH environment variable in JVM profile DFHJVMPR. Also, verify that the CICS paths are included. The actual names of the CICS paths will most likely be different in your environment. Change these appropriately. The last path name in the list should end with :. From the ISPF Command screen, enter:

```
oedit
    Directory ===> /u/edi/cicshome33
    Filename ===> DFHJVMPR

CLASSPATH=\
    /usr/lpp/cicsts/cicsts23/lib/dfjwrap.jar:\
    /usr/lpp/cicsts/cicsts23/lib/dfjcics.jar:\
    /u/edi/cicshome33/edicevh.jar:\
    /u/edi/cicshome33/edicevhsample.jar:\
    /u/edi/cicshome33/wdi.properties:.
```

___ 6. If the WebSphere Data Interchange sample email notification is to be used, then activation.jar and mail.jar must be downloaded from http://www.sun.com. These jars are part of the Sun JavaMail API and Sun Activation Frameworks. Once these jars are downloaded, their paths must be included in the DFHJVMPR class path. See step 5 above for instructions on how to update the CLASSPATH in DFHJVMPR. The last path name in the list should end with :.

___ 7. If WebSphere MQ is to be used, then com.ibm.mq.jar, com.ibm.mqjms.jar, and connector.jar files must also be included in the DFHJVMPR class path. These jar files are located in the IBM

WebSphere MQ product java lib directory. See step 5 above for instructions on how to update the CLASSPATH in DFHJVMPR. The last path name in the list should end with :.

__ 8. If the CICS System Initialization Table (SIT) JVMPROFILEDIR parameter is not already set to the WebSphere Data Interchange CICS home directory /u/edi/cicshome33, then set it now.

```
JVMPROFILEDIR=/u/edi/cicshome33
```

__ 9. Update the wdi.properties file. The default version of this file indicates the Java API soruce is a dataset (wdiSource WDIDataSetSession is uncommented). For the initial WebSphere Data Interchange CICS installation, the Java API source should be transient data queue EDI6. (Here TDQ EDI6 would be specified as a print file destination in a WDI Event Destination profile.) Please comment the source dataset setting, uncomment the source TDQ setting, and set the TDQ name to EDI6. To update wdi.properties, from the ISPF Command screen, enter:

```
oedit
  Directory ===> /u/edi/cicshome33
  Filename ===> wdi.properties

  #wdiSource=com.ibm.edi.wdievents.WDIDirectory2Session
  #wdiSource=com.ibm.edi.wdievents.WDIFile2Session
  #wdiSource=com.ibm.edi.wdievents.WDIQueue2Session
  #wdiSource=com.ibm.edi.wdievents.WDIDataSet2Session
  wdiSource=com.ibm.edi.wdievents.WDITDQ2Session
  #wdiSource=com.ibm.edi.wdievents.WDITSQ2Session

  #WDITDQ2Session properties
  TDQueueName=EDI6
```

Comment the DummyEventHandler wdiHandler setting and uncomment the WDIEmailNotify wdiHandler setting:

```
wdiHandler=com.ibm.edi.wdievents.email.WDIEmailNotify
#wdiHandler=com.ibm.edi.wdievents.email.DummyEventHandler
```

Set the smtpServerAddress property. This should be the URL or IP address of your SMTP (Simple Mail Transfer Protocol) email server. smtpServerAddress is the address of the SMTP server for outgoing messages.   You get this information from your LAN administrator or internet service provider (ISP):

```
smtpServerAddress=nnn.nnn.nnn.nnn
```

Set the FROM and TO addresses of the email notification. The FROM address is the address of the email sent. The TO address is the address to receive the notification. Change these appropriately

```
fromAddress=fromuser@address.com
toAddresses=touser@address.com
```

If your SMTP server requires user ID and password authentication, you may specify these in the userId and password properties. Uncomment these and change the if necessary. Otherwise, leave them commented out:

```
#userId=myUserId
#password=sekret
```

The property values in wdi.properties may later be changed to suit your installation preferences. For example, you may elect to use an MQ rather than a TDQ as your Java API source.

__ 10. Go to 3.2.14, "Step 36. Customizing the CICS Startup JCL for WebSphere Data Interchange."

3.2.14 Step 36. Customizing the CICS Startup JCL for WebSphere Data Interchange

In this step, you customize the CICS startup JCL used to start the CICS region where WebSphere Data Interchange transactions can be invoked.

The following steps describe how to customize the CICS startup JCL.

__ 1. If you will be using WebSphere Data Interchange's export/import function in your CICS environment and the export/import transient data queue is extra-partition queue, EDIEIEXT, then add the following DD statement to your CICS startup JCL. Export/import transient data queue, EDIEIEXT, is the default provided in WebSphere Data Interchange's FXXODCT job. Change the fully-qualified dataset name as appropriate. The dataset attributes must be RECORDSIZE(8152) BLOCKSIZE(8156) variable blocked.

```
//EDIEIEXT DD DSN=CUSTOMER.EDIEIEXT,DISP=SHR
```

__ 2. Add the following data sets to the STEPLIB concatenation:

- Language Environment library:
  – SYS1.SCEERUN
- If using WebSphere MQ Series, libraries:
  – CSQ.SCSQANLE
  – CSQ.SCSQAUTH
  – CSQ.SCSQMVR1

__ 3. Add the following data sets to the DFHRPL concatenation:

- WebSphere Data Interchange load library EDI.V3R3M0.SEDILMD2
- Load library where your CICS tables are link-edited into
- DB2 load library
- C++ IBM Open Class Library
  – SYS1.SCLBDLL
- Language Environment libraries:
  – SYS1.SCEECICS
  – SYS1.SCEERUN
- If using WebSphere MQ Series, libraries:
  – CSQ.SCSQANLE
  – CSQ.SCSQAUTH
  – CSQ.SCSQCICS
  – CSQ.SCSQMVR1
- Network interface load library if any

__ 4. If not already enabled, install CICS support for Language Environment.   See CICS documentation for details.

__ 5. Go to 3.2.15, "Step 37. Verifying the CICS Installation."

3.2.15 Step 37. Verifying the CICS Installation

If you are migrating from a previous version of WebSphere Data Interchange, please review the operational changes discussed in 2.1.22, "Step 20. Operational Changes."

If you are migrating from DataInterchange Version 3.1, see Table 9 on page 37 for a list of control block changes. It may be necessary that user written programs using WebSphere Data Interchange control blocks be recompiled.   In addition, if there were any WebSphere Data Interchange control block field name changes, user written programs that reference those fields will have to be updated before recomplilation.

The Utility Control Information control block (FFUS) changed for WebSphere Data Interchange V3.3. The size of the control block changed from 248 to 300 bytes, and a few new fields were added to the extended part of the block. It is not necessary that user-written WebSphere Data Interchange Utility-invoking programs be recompiled in order to still execute successfully. At offset 211 (position 212) is a new field that indicates whether the old block size is in effect (248) or the new block size is in effect (300). If this field contains an 'X', then the new size is in effect. Your current applications would not have an 'X' in this field, so the old block size is assumed. The new fields that were added to the extended part of the block are for the XML print file name and type, and for the ADF print file name and type. If the defaults for these are acceptable to your WebSphere Data Interchange Utility-invoking programs, then your programs do not need changes and do not need recompiling.

The following steps describe how to verify the WebSphere Data Interchange CICS Installation.

__ 1. Start the CICS region using the startup JCL updated in the previous steps.

__ 2. If you happen to have temporary storage queue EDITV00 set up as being recoverable or if you are not sure if it is recoverable, make sure it is purged. You can see whether the queue exists by entering CEBR EDITV00 from a blank CICS screen. If it does exist, enter PURGE on the CEBR command line. Alternatively, you can enter CEMT SET TSQUEUE(EDITV00) DELETE from a blank CICS screen.

__ 3. From a blank CICS screen, enter transaction EDIV. A successful completion of the transaction verifies the WebSphere Data Interchange CICS installation. There should be no WebSphere Data Interchange-related error messages in the CICS log.

Appendix

Appendix A. Reorganization of WebSphere Data Interchange Repository Data

This section describes how to reorganize your WebSphere Data Interchange DB2 tables. The distribution tape provides the sample JCL.

A.1 Reorganization of Repository DB2 Tables

The following steps describe how to reorganize the repository DB2 tables.

___ 1. Edit and run the JCL member EDIBKUPD in EDI.V3R3M0.SEDIINS1 to back up the repository DB2 tables.

___ 2. Drop all repository DB2 tables and recreate them following the instructions in 2.1.11, "Step 11. Creating Version 3.3 Repository DB2 Objects" on page 27.

___ 3. Edit and run the JCL member EDIRSTRD in EDI.V3R3M0.SEDIINS1 to restore the repository DB2 tables.

___ 4. Verify that all return codes are zero for each step. If you receive a nonzero return code, see the appropriate product's message manual for further information.

Appendix B. Migrating Operational Data From WDI V3.2 to WDI V3.3 on z/OS

If you are currently running WDI 3.2, you can migrate your maps, profiles, and many other objects to your WDI 3.3 database using the import/export or release migration features of WDI. However, certain types of operational data including Transaction Store, Management Reporting, and SAP status data cannot be imported and exported. Special migration programs are included to allow you to migrate this data from your WDI 3.2 database to your WDI 3.3 database.

Some restrictions on the use of these programs:

- These programs only support the migration from WDI 3.2 databases to WDI 3.3 databases. Migration of Transaction Store, Management Reporting, and SAP status data from WDI 3.1 is not supported.
- These programs only support migration to the same type of operating system. For example, you can use them to migrate from a WDI 3.2 database on z/OS to a WDI 3.3 database on z/OS, but not from 3.2 on z/OS to 3.3 on Windows or AIX.
- The WDI 3.2 system that you are migrating from must have PTF PQ78335 applied.
- These programs are intended to **REPLACE ALL** existing Transaction (Document) Store, Management Reporting, and SAP status information in the new database. Although it is possible to modify the JCL to insert records into the WDI 3.3 database without deleting the existing records first, this should only be used in special situations, and is generally not recommended. If the migrated records conflict with records that were generated by WDI 3.3 processing, it can put the database into an invalid state, and later cause problems during translation and/or reporting functions. To avoid this situation, the JCL to load the data will use the "clean" option to delete all existing records from the WDI 3.3 table before it loads the migrated data.
- There are a number of interdependencies between Transaction Store tables, and also between Management Reporting tables. You should migrate these as a set, and not try to migrate individual tables. Migrating individual tables can put the database into an invalid state, and later cause problems during translation and/or reporting functions.

These migration steps can be done either immediately after you create your WDI 3.3 database and load the default data, or they can be done later after you have done additional setup and testing with WDI 3.3. If you do the migration after you have done additional testing with WDI 3.3, **remember that any existing WDI 3.3 Transaction Store, Management Reporting, and SAP status data will be deleted during the migration of the WDI 3.2 data**. Other data such as maps, profiles, etc. will not be affected by these migration steps.

The steps to migrate the operational data follow:

B.1 Step 1. Binding the DB2 Plan for unload processing

This step uses the IKJEFT01 utility to bind the WebSphere Data Interchange EDIPUNL DB2 application plan for use in TSO and batch. This plan is used only by the EDIMUNL utility to unload select tables from the WDI 3.2 database.

The following steps describe how to BIND the DB2 Plan:

__ 1. Edit the EDIJBUNL member in EDI.V3R3M0.SEDIINS1. Modify the member for your environment as shown in the comment at the beginning of the JCL.

__ 2. Run the job.

__ 3. Verify that the return code is zero. If you receive a return code greater than zero, see the appropriate product's message manual for further information.

B.2 Step 2. Granting Permission on the DB2 Plan for unload Processing

After binding the EDIPUNL DB2 application plan, you must make it available to the user(s) that will run the EDIMUNL utility in Step 3 to unload the WDI 3.2 data. The permissions for this plan may be different from other plans defined for WDI 3.2 or WDI 3.3.

Note: SQL member EDISGUNL of the PDS EDI.V3R3M0.SEDISQL1 provides sample SQL statements for granting authorization on the EDIPUNL plan.

__ 1. You must authorize GRANT EXECUTE authority to the user that will run the utility to unload the data in Step 3. You can use SPUFI to run an SQL statement similar to the following example that makes plan EDIPUNL available to user ID USER01:

```
GRANT EXECUTE ON PLAN EDIPUNL TO USER01;
```

__ 2. Verify that the return code is zero. If you receive a return code greater than zero, see the appropriate product's message manual for further information.

B.3 Step 3. Unloading the WDI V3.2 Operational Data From WDI V3.2

EDIJMUNL runs the WDI 3.3 utility program, EDIMUNL, to unload specific tables from the WDI 3.2 database. These tables contain operational data including Transaction Store, Management Reporting, and SAP status data. The data is written to datasets using the high-level qualifier EDI.MIG32. A separate file is created for each table.

The following steps describe how to unload the operational data from the WDI 3.2 database.

__ 1. Edit the EDIJMUNL member in EDI.V3R3M0.SEDIINS1. Modify the member for your environment as shown in the comment at the beginning of the JCL.

__ 2. Run the job.

___ 3. Verify that all return codes are zero. If you receive a nonzero return code, see the appropriate product's message manual for further information.

B.4 Step 4. Binding the DB2 Plan for Load Processing

This step uses the IKJEFT01 utility to bind the WebSphere Data Interchange EDIPLOD DB2 application plan for use in TSO and batch. This plan is used only by the EDIMLOD utility to load the migrated data into the WDI 3.3 database.

The following steps describe how to bind the DB2 plan:

___ 1. Edit the EDIJBLOD member in EDI.V3R3M0.SEDIINS1. Modify the member for your environment as shown in the comment at the beginning of the JCL.

___ 2. Run the job.

___ 3. Verify that the return code is zero. If you receive a return code greater than zero, see the appropriate product's message manual for further information.

B.5 Step 5. Granting Permission on the DB2 Plan for Load Processing

After binding the EDIPLOD DB2 application plan, you must make it available to the user(s) that will run the EDIMLOD utility in Step 6 to load the data into WDI 3.3. The permissions for this plan may be different from other plans defined for WDI 3.2 or WDI 3.3.

Note: SQL member EDISGLOD of the PDS EDI.V3R3M0.SEDISQL1 provides sample SQL statements for granting authorization on the EDIPLOD plan.

___ 1. You must authorize GRANT EXECUTE authority to the user that will run the utility to load the data in Step 6. You can use SPUFI to run an SQL statement similar to the following example that makes plan EDIPLOD available to user ID USER01:

```
GRANT EXECUTE ON PLAN EDIPLOD TO USER01;
```

___ 2. Verify that the return code is zero. If you receive a return code greater than zero, see the appropriate product's message manual for further information.

B.6 Step 6. Loading the WDI 3.2 operational data into WDI V3 .3

EDIJMLOD runs the WDI 3.3 utility program, EDIMLOD, to load the data into the WDI 3.3 database from the files created in the previous steps. These files contain operational data including Transaction Store, Management Reporting, and SAP status data. The data is read from datasets using the high-level qualifier EDI.MIG32. A separate file is used for each table.

Note: This step will delete existing Transaction Store, Management Reporting, and SAP status data from the WDI 3.3 database before loading the migrated data.

The edimlod program will do any necessary conversions on the WDI 3.2 records so it can insert the data into the WDI 3.3 database. Some notes on these conversions:

- Since the WDI 3.2 database did not support Unicode, data is assumed to be in the system default codepage. Values are converted to Unicode during the load process as required.
- The sender and receiver qualifier values are included in the Transaction Store tables for WDI 3.3, but were not saved in WDI 3.2. For records that are migrated from WDI 3.2, these will be set to a value of "*". If you use the qualifiers as part of your filter criteria for queries and extracts, you will need to specify an "*" to see the migrate data. Or, you can omit the qualifier from your filter criteria.
- When WDI server updates the Transaction Store status to reflect received functional acknowledgements or network communications, it has special logic to check for entries with the "*" qualifiers if it is unable to find the Transaction Store entries that match the specific qualifiers in the data.

The following steps describe how to load the migrated operational data into the WDI 3.3 database.

___ 1. Edit the EDIJMLOD member in EDI.V3R3M0.SEDIINS1. Modify the member for your environment as shown in the comment at the beginning of the JCL.

___ 2. Run the job.

___ 3. Verify that all return codes are zero. If you receive a nonzero return code, see the appropriate product's message manual for further information.

Notices

This information was developed for products and services offered in the U.S.A.

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 that IBM product, program, or service may be used. Any functionally equivalent product, program, or service that does not infringe any IBM intellectual property right 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 grant you any license to these patents. You can send license inquiries, in writing, to:

IBM® Director of Licensing
IBM Corporation
North Castle Drive
Armonk, N.Y. 10504-1785
U.S.A.

For license inquiries regarding double-byte (DBCS) information, contact the IBM Intellectual Property Department in your country or send inquiries, in writing, to:

IBM World Trade Asia Corporation Licensing
2-31 Roppongi 3-chome, Minato-ku
Tokyo 106-0032, Japan.

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 information 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.

Any references in this information to non-IBM Web sites are provided for convenience only and do not in any manner serve as an endorsement of those Web sites. The materials at those Web sites are not part of the materials for this IBM product and use of those Web sites is at your own risk.

IBM may use or distribute any of the information you supply in any way it believes appropriate without incurring any obligation to you.

Licensees of this program who wish to have information about it for the purpose of enabling: (i) the exchange of information between independently created programs and other programs (including this one) and (ii) the mutual use of the information which has been exchanged, should contact:

IBM Corporation
Department DD40
P.O. Box 30021
Tampa, Florida 33630-3021 USA

Such information may be available, subject to appropriate terms and conditions, including in some cases, payment of a fee.

The licensed program described in this document and all licensed material available for it are provided by IBM under terms of the IBM Customer Agreement, IBM International Program License Agreement or any equivalent agreement between us.

Any performance data contained herein was determined in a controlled environment. Therefore, the results obtained in other operating environments may vary significantly. Some measurements may have been made on development-level systems and there is no guarantee that these measurements will be the same on generally available systems. Furthermore, some measurements may have been estimated through extrapolation. Actual results may vary. Users of this document should verify the applicable data for their specific environment.

Information concerning non-IBM products was obtained from the suppliers of those products, their published announcements or other publicly available sources. IBM has not tested those products and cannot confirm the accuracy of performance, compatibility or any other claims related to non-IBM products. Questions on the capabilities of non-IBM products should be addressed to the suppliers of those products.

All statements regarding IBM's future direction or intent are subject to change or withdrawal without notice, and represent goals and objectives only.

All IBM prices shown are IBM's suggested retail prices, are current and are subject to change without notice. Dealer prices may vary.

This information is for planning purposes only. The information herein is subject to change before the products described become available.

This information contains examples of data and reports used in daily business operations. To illustrate them as completely as possible, the examples include the names of individuals, companies, brands, and products. All of these names are fictitious and any similarity to the names and addresses used by an actual business enterprise is entirely coincidental.

COPYRIGHT LICENSE

This information contains sample application programs in source language, which illustrate programming techniques on various operating platforms. You may copy, modify, and distribute these sample programs in any form without payment to IBM, for the purposes of developing, using, marketing or distributing application programs conforming to the application programming interface for the operating platform for which the sample programs are written. These examples have not been thoroughly tested under all conditions. IBM, therefore, cannot guarantee or imply reliability, serviceability, or function of these programs.

If you are viewing this information softcopy, the photographs and color illustrations may not appear.

Programming interface information

Programming interface information, if provided, is intended to help you create application software using this program.

General-use programming interfaces allow you to write application software that obtain the services of this program's tools.

However, this information may also contain diagnosis, modification, and tuning information. Diagnosis, modification and tuning information is provided to help you debug your application software.

Warning: Do not use this diagnosis, modification, and tuning information as a programming interface because it is subject to change.

Trademarks and service marks

The following terms are trademarks or registered trademarks of International Business Machines Corporation in the United States or other countries, or both:

IBM
the IBM logo
AIX
CICS
IBMLink
IMS
WebSphere MQ
MVS
OS/390
WebSphere
z/OS

Microsoft, Windows, Windows NT, and the Windows logo are trademarks of Microsoft Corporation in the United States, other countries, or both.

MMX, Pentium, and ProShare are trademarks or registered trademarks of Intel Corporation in the United States, other countries, or both.

Java and all Java-based trademarks are trademarks of Sun Microsystems, Inc. in the United States, other countries, or both.

Linux is a trademark of Linus Torvalds in the United States, other countries, or both.

Other company, product or service names may be trademarks or service marks of others.

Glossary of terms and abbreviations

This glossary defines WebSphere Data Interchange terms and abbreviations used in this book. If you do not find the term you are looking for, see the index or the *IBM Dictionary of Computing*, New York: McGraw-Hill, 1994.

This glossary includes terms and definitions from the *American National Dictionary for Information Systems*, ANSI X3.172-1990, copyright 1990 by the American National Standards Institute. Copies may be ordered from the American National Standards Institute, 11 West 42 Street, New York, New York 10036. Definitions are identified by the symbol (A) after the definition.

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*.

ADF. *See 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 WebSphere Data Interchange is an enabled application.

application data. The actual data in an application data file.

application data format. See *data format*.

application default profile. Identifies business applications, such as purchasing and accounts receivable, to WebSphere Data Interchange and sets specific WebSphere Data Interchange processing defaults for an application.

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.

CD-ROM. Compact Disk-Read Only Memory; a storage medium for large amounts of data needed external to the personal computer.

client-server. A computing environment in which two or more machines work together to achieve a common task.

code list. A table, supplied by WebSphere Data Interchange or defined by the user, that contains all acceptable values for a single data field.

composite data element. In EDI standards, a group of related subelements, such as the elements that make up a name and address.

compound element. An item in the source or target document that contains child items. Examples are EDI segments and composite data elements, data format records and structures, and XML elements.

control number. Numbers (or masks used to create numbers) that are used to identify an interchange, group, or EDI transaction.

control string. An object compiled from a map, data format, and EDI standard transaction; it contains the instructions used by the translator to translate a document from one format to another.

Glossary

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

data dictionary. A file containing the definitions of all the data elements of an EDI standard.

data element. A single item of data in an EDI 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 an EDI standard.

data format. A description of the application data for a particular transaction. A data format is composed of loops, records, data structures, and fields.

data format dictionary. A file that contains data format components.

data format record. A group of logically related fields set up as a record in a data format.

data format 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 composite data element in an EDI standard.

DataInterchange/MVS™. The IBM DataInterchange product used on the host; pieces include a TSO parameter entry mechanism and a

translator. The functionality available in this product is now available in WebSphere Data Interchange for z/OS.

DataInterchange/MVS-CICS. The CICS-based IBM DataInterchange product. The functionality available in this product is now available in WebSphere Data Interchange for z/OS.

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.

data transformation map. One of three supported map types. A data transformation map is a set of mapping instructions that describes how to translate data from a source document into a target document. Both the source and target documents can be one of several support document types.

DB2®. Database 2, an IBM relational database management system.

ddname. Data definition name.

decimal notation. The character that represents a decimal point in the data.

delimiter. A character that terminates a string of characters, such as the value contained in a data element.

DI Client. WebSphere Data Interchange Client; the Windows-based, client/server interface for WebSphere Data Interchange.

dictionary. See *data dictionary*.

document. A business document that is exchanged between two enterprises as part of a business process, such as a purchase order or invoice. A document within WebSphere Data Interchange is singular. For example, it cannot contain multiple purchase orders. A document can also be represented in any syntax. For example, an XML purchase order and an EDI purchase order are both documents.

Document Type Definition (DTD). A list of all components included in the XML document and

their relationship to each other. This defines the structure of an XML document.

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.

DTD. See *Document Type Definition*.

E

EDI. Electronic data interchange.

EDIA. Electronic Data Interchange Association.

EDI administrator. The person responsible for setting up and maintaining WebSphere Data Interchange.

EDI message. See message.

EDI standard. The industry-supplied, national, or international formats to which information is converted, allowing different computer systems and applications to interchange information.

EDI transaction. A single business document, such as an invoice.

EDI transaction set. A group of logically related data that make up an electronic business document, such as an invoice or purchase order.

EDIFACT. Electronic Data Interchange for Administration Commerce and Transport. See UN/EDIFACT.

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 tasks, such as a software error, sending a transaction, or acknowledging a message.

Extensible Markup Language (XML). A standard metalanguage for defining markup languages that was derived from, and is a subset of SGML. It is used to represent structured documents and data.

F

field. See *data field*.

floating segment. A segment of an EDI standard that may exist in many positions relative to other segments.

forward 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. Forward translation tables translate local values to standard values.

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.

Glossary

G

global variable.. A variable that is shared among all instances of all documents within a translation session.

H

header. A control structure that indicates the start of an electronic transmission.

hierarchical loop. A technique for describing the relationship of data entities which are related in a parent/child manner, like a corporate organization chart. Used in mapping to group related data elements and segments such as trading partner address.

HL. *See hierarchical loop*.

I

IBM Global Network. The IBM communications network that provides products and services to IBM customers.

ICS. International Control Segments.

import. The process of taking WebSphere Data Interchange objects exported on another WebSphere Data Interchange system and incorporating them into the receiving system.

Information Exchange. A commerce engine of IBM Interchange Services for e-business that permits users to send and receive information electronically.

interchange. The exchange of information between trading partners.

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

literal. In mapping, a value that is constant for each occurrence of the translation. If you provide the literal value during mapping, the translator does not have to refer repeatedly to the source to obtain the value.

local variable. A variable that is specific to the instance of the document in which it is being used.

log file. A file in which events are recorded.

logging. The recording of events in time sequence.

loop. A repeating group of related segments in a transaction set or a repeating group of related records and loops in a data format.

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 a transaction set.

M

mailbox. If you use a mail type protocol to exchange messages with your trading partners, you will have one or more registered mailboxes. The mailbox profile is used in WebSphere Data Interchange to define your mailboxes and any associated preferences.

map. A set of instructions that indicate to WebSphere Data Interchange how to translate data from one format to another.

map rule. An association between a data transformation map and a trading partner.

maximum use. A number indicating the maximum number of times a segment can be used in a transaction set or the maximum number of times that a data format loop or record can repeat.

message. A free-form, usually short, communication to a trading partner. In UN/EDIFACT standards, a group of logically related data that make up an electronic business document, such as an invoice. A message is equivalent to a document.

message log. The file in which WebSphere Data Interchange Client logs messages about errors that occur within the client.

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.

N

network acknowledgment. A response from the network indicating the status of an interchange envelope, such as sent or received.

network commands. The commands that you want WebSphere Data Interchange to pass to your network, defined in the network commands profile. In the host product, this file is named NETOP.

O

ODETTE. Organization for Data Exchange through Teletransmission in Europe.

P

parse. To break down into component parts.

path qualified mapping. A form of mapping in which all occurrences of a repeating compound or simple data element are mapped to a repeating compound or simple data element in another document.

PDS. Partitioned data set.

PDS members. Groups of related information stored in partitioned data sets.

profile. Descriptive information about trading partners, network connections, and so on. Each profile can contain one or more objects or members. For example, the trading partner profile contains members for your trading partners (one member for trading partner address).

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.

R

receive map. One of three supported map types. A receive map is a set of mapping instructions that describe how to translate an EDI standard transaction into a proprietary application data document.

receive usage. An association between a receive map and a trading partner.

record. A logical grouping of related data structures and fields.

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/usages, that WebSphere Data Interchange uses to process your data.

requestor. See *mailbox*.

reverse 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. Reverse translation tables translate standard values to local values.

rule. See *map rule*.

runtime data. Data used by the WebSphere Data Interchange translator, such as control strings, code lists, translation tables and profiles.

S

security administrator. The person who controls access to business data and program functions.

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 an EDI 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.

send map. On of three supported map types. A send map is a set of mapping instructions that describe how to translate a proprietary application data document into an EDI standard transaction.

send usage. An association between a send map and a trading partner.

simple element. An item in the source or target document that does not contain child items, only data. Examples are EDI data elements, data format fields, XML attributes, and PCDATA values.

single-occurrence mapping. A form of mapping in which each occurrence of a loop or repeating compound or simple data element in a document is mapped to a different compound or simple data element in another document.

source document definition. A description of the document layout that will be used to identify the format of the input document for a translation.

special literal. The send and receive Mapping Data Element Editors include the Literal or Mapping Command field. Literals are constant values you enter in this field, such as 123. Special literals are values you enter in this field that begin with an ampersand (&) and are command to WebSphere Data Interchange, rather than constant values. For example, to use today's date, you enter &DATE.

standards. See *EDI standard*.

structure. See *data structure* or *data format structure*.

subelement. In UN/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.

T

tag. In UN/EDIFACT standards, the segment identifier. In export/import, a code identifies each field in the export record. Such export/import files are known as "tagged" files.

target document definition. A description of the document layout that will be used to create an output document from a translation.

TD queue. See *transient data queue*.

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.

TPT. Trading partner transaction. See *map*.

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. See *map*.

trailer. A control structure that indicates the end of an electronic transmission.

transaction. A single business document, such as an invoice. See also *EDI transaction*.

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. See also *EDI transaction set*.

Transaction Store. The file that contains the results of translations and a history of translation activity.

transform. The process of converting a document from one format to another.

transient data queue (TD). A sequential data set used by the Folder Application Facility in CICS to log system messages.

translation. The process of converting a document from one format to another.

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.

TSQ. See *temporary storage queue*.

U

UCS. Uniform Communication Standard.

unary operator. An operator that changes the sign of a numeric value.

UN/EDIFACT. United Nations Electronic Data Interchange for Administration Commerce and Transport.

Uniform Communication Standard (UCS). The EDI standard used in the grocery industry.

UN/TDI. United Nations Trade Data Interchange.

Usage. An association between a send or receive map and a trading partner.

V

validation table. A table, supplied by WebSphere Data Interchange or defined by the user, which contains all acceptable values for a single data field.

variable. The entity in which a value may be stored based on data received; as opposed to a constant value.

W

WebSphere Data Interchange. A generic term for the WebSphere Data Interchange products, WebSphere Data Interchange for z/OS and WebSphere Data Interchange for Multiplatforms. WebSphere Data Interchange is a translator of data from one document format to another; the pieces of this product include a TSO parameter entry mechanism, a CICS parameter entry mechanism, a Windows-based parameter entry mechanism (WebSphere Data Interchange Client), and a translator.

WebSphere Data Interchange Client. A Windows-based product for entry of parameters needed by the WebSphere Data Interchange translator.

Glossary

WebSphere MQ. An IBM product that is used to implement messaging and queueing of data groups. Earlier releases of this product were known as MQSeries®.

WebSphere MQ queue profile. Represents a relationship between a logical name and a physical WebSphere MQ queue name.

WINS. Warehouse Information Network Standard.

Windows®. Microsoft's graphical operating system under which WebSphere Data Interchange Client runs.

X

X12. A common EDI standard approved by the American National Standards Institute.

XML. See *Extensible Markup Language*.

Bibliography

This section describes the documentation available for the WebSphere Data Interchange product.

WebSphere Data Interchange publications

The WebSphere Data Interchange V3.3 publications are:

- *WebSphere Data Interchange for MultiPlatforms Quick Start Guide* CF0YREN
- *WebSphere Data Interchange for MultiPlatforms Administration and Security Guide* SC34-6214-01
- *WebSphere Data Interchange for MultiPlatforms Messages and Codes* SC34-6216-01
- *WebSphere Data Interchange for MultiPlatforms User's Guide* SC34-6215-01
- *WebSphere Data Interchange for MultiPlatforms Programmer's Reference* SC34-6217-01
- *WebSphere Data Interchange for MultiPlatforms Mapping Guide* SC23-5874-00
- *WebSphere Data Interchange for MultiPlatforms Utility Commands and File Formats Reference* SC23-5873-00
- *WebSphere Data Interchange for z/OS V3.3 Program Directory* GI10-2561-01
- *WebSphere Data Interchange for z/OS V3.3 Installation Guide* SC34-6269-01
- *WebSphere Data Interchange for z/OS V3.3 License File* GC34-6270-02

Softcopy books

All the WebSphere Data Interchange books are available in softcopy format.

Portable Document Format (PDF)

The library is supplied as stand-alone PDFs in US English in the DOC directory on the product CD. The contents of the DOC directory can be viewed without installing the product.

PDF files can be viewed and printed using the Adobe Acrobat Reader. You will need Adobe Acrobat Reader with Search Version 4.05 on Windows NT, or Adobe Acrobat Reader with Search Version 4.5 on UNIX® systems.

If you need to obtain the Adobe Acrobat Reader, or would like up-to-date information about the platforms on which the Acrobat Reader is supported, visit the Adobe Systems Inc. Web site at:

```
http://www.adobe.com/
```

If you cut and paste examples of commands from PDF files to a command line for execution, you must check that the content is correct before you press Enter. Some characters might be corrupted by local system and font settings.

WebSphere Data Interchange information available on the Internet

The WebSphere Data Interchange product Web site is at:

```
http://www.ibm.com/websphere/datainterchange/
```

By following links from this Web site you can:

- Obtain latest information about the WebSphere Data Interchange products.
- Access the WebSphere Data Interchange books in PDF format.

Index

IBM ®

Printed in USA