



Informatica® PowerExchange
10.4.0

Reference Manual

© Copyright Informatica LLC 2005, 2021

This software and documentation are provided only under a separate license agreement containing restrictions on use and disclosure. No part of this document may be reproduced or transmitted in any form, by any means (electronic, photocopying, recording or otherwise) without prior consent of Informatica LLC.

U.S. GOVERNMENT RIGHTS Programs, software, databases, and related documentation and technical data delivered to U.S. Government customers are "commercial computer software" or "commercial technical data" pursuant to the applicable Federal Acquisition Regulation and agency-specific supplemental regulations. As such, the use, duplication, disclosure, modification, and adaptation is subject to the restrictions and license terms set forth in the applicable Government contract, and, to the extent applicable by the terms of the Government contract, the additional rights set forth in FAR 52.227-19, Commercial Computer Software License.

Informatica, the Informatica logo, PowerCenter, and PowerExchange are trademarks or registered trademarks of Informatica LLC in the United States and many jurisdictions throughout the world. A current list of Informatica trademarks is available on the web at <https://www.informatica.com/trademarks.html>. Other company and product names may be trade names or trademarks of their respective owners.

Portions of this software and/or documentation are subject to copyright held by third parties. Required third party notices are included with the product.

The information in this documentation is subject to change without notice. If you find any problems in this documentation, report them to us at infa_documentation@informatica.com.

Informatica products are warranted according to the terms and conditions of the agreements under which they are provided. INFORMATICA PROVIDES THE INFORMATION IN THIS DOCUMENT "AS IS" WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING WITHOUT ANY WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND ANY WARRANTY OR CONDITION OF NON-INFRINGEMENT.

Publication Date: 2021-04-08

Table of Contents

Preface	14
Informatica Resources.	14
Informatica Network.	14
Informatica Knowledge Base.	14
Informatica Documentation.	14
Informatica Product Availability Matrices.	15
Informatica Velocity.	15
Informatica Marketplace.	15
Informatica Global Customer Support.	15
 Chapter 1: Introduction to PowerExchange.....	16
PowerExchange Overview.	16
PowerExchange Components.	16
 Chapter 2: DBMOVER Configuration File.....	19
DBMOVER Configuration File Overview.	19
DBMOVER Statement Summary Tables.	19
Summary of All DBMOVER Statements	20
Summary of Data Source-Specific DBMOVER Statements.	34
Summary of Operating System-Specific DBMOVER Statements.	42
Syntax of DBMOVER Statements.	49
Syntax Rules and Guidelines for DBMOVER Statements.	49
Notational Conventions for DBMOVER Statements.	50
DBMOVER Statements.	50
ABEND_SW Statement.	50
ADA_L3_ALLOW Statement.	51
ADA_TIMEZONE_CHECK Statement.	52
ADABAS_DEFAULT_DBID Statement.	52
ADABAS_MU_SEARCH Statement.	53
ADABAS_PREFIX Statement.	54
ADABASCODEPAGE Statement.	54
ADAOPT Statement.	55
ADAOPTM Statement.	56
ADAPREFETCH Statement.	56
ADASTATS Statement.	56
ADAUSER Statement.	57
APPBUFSIZE Statement.	57
APPBUFSIZEDYN Statement.	58
AS400EVENTMSGQ Statement.	58
AS400USRJRNCODE Statement.	59

AS400USRMSGQ Statement.	60
BS Statement.	61
CAPI_CONN_NAME Statement.	61
CAPI_CONNECTION - AS4J Statement.	62
CAPI_CONNECTION - CAPX Statement.	67
CAPI_CONNECTION - LRAP Statement.	69
CAPI_CONNECTION - MSQL Statement.	71
CAPI_CONNECTION - MYSQL Statement.	77
CAPI_CONNECTION - ORAD Statement.	80
CAPI_CONNECTION - PG Statement.	82
CAPI_CONNECTION - UDB Statement.	85
CAPI_CONNECTION - UOWC Statement.	89
CAPI_SRC_DFLT Statement.	94
CAPT_PATH Statement.	96
CAPT_XTRA Statement.	97
CMDNODE Statement.	97
CODEPAGE Statement.	98
COLON Statement.	99
COMPRESS Statement.	99
CONSOLE_CODEPAGE Statement.	100
CONSOLE_MSG Statement.	100
CONSOLE_TRACE Statement.	101
CONVCHAR Statement.	101
CPX_DIR Statement.	102
CREDENTIALS_CASE Statement.	102
DATAMAP_SERVER Statement.	103
DATERANGE Statement.	104
DB2_BIN_AS_CHAR Statement.	105
DB2_BIN_CODEPAGE Statement.	105
DB2_ERRORFILE Statement.	106
DB2CODEPAGE Statement.	106
DB2DEF_ENCODING Statement.	110
DB2ID Statement.	111
DB2PLAN Statement.	112
DECPOINT Statement.	112
DEFAULTCHAR Statement.	113
DEFAULTDATE Statement.	113
DISABLE_PARTITIONS Statement.	113
DISP Statement.	114
DM_RESOURCE Statement.	115
DM_SUBTASK Statement.	115
DMXCACHE_DELETEECSA Statement.	116

DMXCACHE_MAX_MEMORY_MB Statement.	117
DMXCACHE_MULTIPLEJOBS Statement.	117
DMX_DIR Statement.	118
DTLMSG_CODEPAGE Statement.	119
DTLMSG_LANGUAGE Statement.	119
ENABLE_AUTOCOMMIT Statement.	120
ENCRYPT Statement.	120
ENCRYPTLEVEL Statement.	121
ENQMAJORNAME Statement.	122
ENQSYSTEMS Statement.	122
ERRROWNOTFOUND Statement.	123
EXT_CP_SUPPT Statement.	123
FILEMAPPED_MEMORY_DIR Statement.	124
GDGLOCATE Statement.	125
GSBUFSIZE Statement.	126
HOSTNAME Statement.	126
ICUALIAS Statement.	127
ICUCNVPROPERTY Statement.	127
ICUCONVERTER Statement.	130
ICUDATADIR Statement.	131
IMSBSDS Statement.	132
IMSID Statement.	133
JOBCLASS Statement.	134
LDAP_BASE Statement.	134
LDAP_BIND_DN Statement.	134
LDAP_BIND_EPWD Statement.	135
LDAP_BIND_PWD Statement.	136
LDAP_BIND_TIMEOUT Statement.	136
LDAP_FILTER Statement.	137
LDAP_HOST Statement.	137
LDAP_LOGIN_ATTRIBUTE Statement.	138
LDAP_OPENSSL Statement.	138
LDAP_PORT Statement.	139
LDAP_SASL_MECH Statement.	140
LDAP_SCOPE Statement.	140
LDAP_SEARCH_TIMEOUT Statement.	141
LDAP_TLS Statement.	141
LISTENER Statement.	142
LOADCTLFILE Statement.	144
LOADJOBFILE Statement.	145
LOG_CODEPAGE Statement.	145
LOG_LINE_LIMIT Statement.	146

LOGPATH Statement.	146
LOGSID Statement.	147
LOWVALUES Statement.	147
LRECL Statement.	148
LU00FILE Statement.	148
MAXTASKS Statement.	149
MSSQL_SERVER_CONNECT_TIMEOUT Statement.	149
MSSQL_SERVER_STATEMENT_TIMEOUT Statement.	150
MSGPREFIX Statement.	150
MSGPREFIX-HYPHEN Statement.	150
MSS_ERRORFILE Statement.	151
MVSDB2AF Statement.	151
NEGSIGN Statement.	152
NETPORT Statement.	152
NODE Statement.	155
NOGETHOSTBYNAME Statement.	157
NRDB_WRITE_CHAR_NULL_FILL Statement.	157
NRDB_WRITE_NUM_NULL_FILL Statement.	157
NUMERICSIGN Statement.	158
ODBASUPP Statement.	158
ORA_ERRORFILE Statement.	159
ORACLE_CAPTURE_TYPE Statement.	159
ORACLE_UNHANDLED_NUMASCHAR Statement.	160
ORACLECODEPAGE Statement.	161
ORACLEID Statement.	162
OUSP Statement.	164
PC_AUTH Statement.	164
PIPE Statement.	165
POLLTIME Statement.	165
PRE861_COMPAT Statement.	166
PRGIND Statement.	166
PRGINT Statement.	167
PWXSOMAXCONN Statement.	167
RACF_CLASS Statement.	168
RDBMSINSRTDFLT Statement.	168
RECFM Statement.	169
REJECT_FILE_DELIMITER Statement.	169
RELEASE Statement.	170
RMTRDBDIRE Statement.	170
RMTSYSNAME Statement.	170
SECURITY Statement.	171
SECURITY_MSGSUPP Statement.	175

SERVICE_TIMEOUT Statement.	175
SESSID Statement.	176
SHOW_THREAD_PERF Statement.	176
SPACE Statement.	177
SSL Statement.	177
SSL_ALLOW_SELFSIGNED Statement.	178
SSL_CIPHER_LIST Statement.	179
SSL_CONTEXT_METHOD Statement.	180
SSL_REQ_CLNT_CERT Statement.	180
SSL_REQ_SRVR_CERT Statement.	181
START_UP_USER_EXIT Statement.	181
STATS Statement.	182
SUBMITTIMEOUT Statement.	184
SUP_FUNC Statement.	184
SUP_REQUEST Statement.	185
SUP_SSNAME Statement.	185
SUP_SSTYPE Statement.	185
SUPPRESS_DATA_LOGGING Statement.	186
SVCNODE Statement.	186
SYSOUT_TIMESTAMP Statement.	188
TAPEWAIT Statement.	188
TCPIP_ACCEPT_NONBLOCK Statement.	188
TCPIP_DIAGNOSTICS_TRACE Statement.	189
TCPIP_DIAGNOSTICS_TRACE_SZ Statement.	189
TCPIP_SHOW_POOLING Statement.	190
TCPIPBUFFSIZE Statement.	191
TCPIPVER Statement.	191
TEMPHLQ Statement.	191
TEXT_EOF_FOR_BINARY Statement.	192
TIMEZONE Statement.	192
TRACE Statement.	193
TRACING Statement.	193
UNIT Statement.	199
USE_DB_AUTH Statement.	199
USE_TYPE1_FALLBACKS Statement.	200
USESUP Statement.	200
VOLSER Statement.	201
VSAM Statement.	201
WAITDSN Statement.	202
WORKCLASS Statement.	202
WRT_ERROR_HANDLING Statement.	203
Consumer API (CAPI) Connection Statements.	204

Multiple CAPI Connections and Overrides.	204
Source-Specific CAPI_CONNECTION Statements.	205
DBMOVER Statements for Commands Issued through the pwxcmd Program.	206
Sample DBMOVER Configuration Files.	207
Chapter 3: Netport Jobs.	209
Netport Jobs Overview.	209
Configuring Netport Jobs.	210
Sample Netport Jobs.	210
Job Status Messages for Netport Jobs.	211
PowerExchange Substitution Variables for Netport Jobs.	212
Example of IMS SSID and PSB Substitution for an IMS Netport Job.	215
Netport Jobs and Offload Processing.	216
Netport Jobs and DB2 Bulk Data Loads.	216
Using Netport Jobs for Generation Data Sets.	216
Setting Up Netport Jobs for GDGs.	217
Chapter 4: PowerExchange Message Logs and Destination Overrides.	218
PowerExchange Message Logs.	218
PowerExchange Message Log File.	219
PowerExchange Alternative Log Files.	219
PowerExchange Message Destination Overrides.	223
DTLMSGO File and Statements.	224
Chapter 5: SMF Statistics Logging and Reporting.	227
SMF Statistics Logging and Reporting Overview.	227
Configuring Statistics Logging to SMF.	228
Configuring Statistics Logging to a PowerExchange File.	229
Generating Reports of SMF Statistics.	230
Defining the Set of SMF Data Sets to Use for Reporting.	231
Sample JCL for Generating Reports.	231
Example Reports.	232
Viewing Statistics Written to an SMF Data Set from the PowerExchange Navigator.	235
Viewing Statistics Written to a PowerExchange File from the PowerExchange Navigator.	239
SMF Record Format.	240
Standard SMF Header with Subtype.	240
SMF Triplet Section Descriptor.	241
PowerExchange Header/Descriptor.	242
Extended Section Descriptor.	242
General Section.	243
Extended Section for the PowerExchange Listener.	244
Extended Section for the PowerExchange Client.	245
Extended Section for Data Access.	246

Function/Access Method Section (DB2).	246
SMF Record Layout.	247
PowerExchange File Record Layout.	248
Chapter 6: PowerExchange Security.....	250
PowerExchange Security Overview.	250
SECURITY Statement.	251
i5/OS Security.	255
Resource Access Requirements on i5/OS.	256
Authorization to Run PowerExchange Listener Commands on i5/OS.	258
User Authentication for the pwxcmd Program on i5/OS.	259
Authorization to Run pwxcmd Commands on i5/OS.	259
Linux, UNIX, and Windows Security.	260
Authorization to Run PowerExchange Listener Commands on Windows.	261
Authorization to Run pwxcmd and infacmd pwx Commands on Linux, UNIX, and Windows. . .	261
User Authentication for the pwxcmd or infacmd pwx Programs on Linux, UNIX, and Windows	262
z/OS Security.	263
Resource Access Requirements on z/OS.	264
Security for Capture Registrations and Extraction Maps on z/OS.	269
Security for Data Maps	273
Security for PowerExchange Listener Commands.	274
Adabas Security.	274
Datacom Security.	275
DB2 for z/OS Security.	276
IMS Security.	277
User Authentication for the pwxcmd Program on z/OS.	277
Authorization to Run pwxcmd Commands on z/OS.	278
PowerExchange Passphrases.	279
PowerExchange LDAP User Authentication.	280
PowerExchange LDAP Authentication Components.	281
Supported Environments for LDAP Authentication.	282
LDAP Search User.	283
Controlling the LDAP Search.	283
LDAP Transport Layer Security.	284
LDAP Server Failover.	285
Configuration Tasks.	286
Example DBMOVER LDAP Configurations.	288
PowerExchange Selective Sign-on.	293
Configuring PowerExchange Selective Sign-on.	293
PowerExchange Sign-on File.	294
PowerExchange Sign-on File Syntax.	294
PowerExchange Sign-on File Statements.	295
PowerExchange Sign-on File Examples.	298

Chapter 7: Secure Sockets Layer Support.....	300
SSL Security on a PowerExchange Network.	300
PowerExchange SSL Architecture.	301
FIPS 140-2 Compliance.	301
FIPS 140-2 Compliant Cipher Suites.	302
FIPS 140-2 Compliance Considerations on z/OS.	303
PWXUGSK Utility.	303
PWXUSSL Utility.	303
PowerExchange SSL Configuration Steps.	304
SSL Server Configuration on z/OS.	305
Updating the AT-TLS Policy File.	305
Creating a Personal Certificate on z/OS.	308
Configuring the PowerExchange Listener on the z/OS SSL Server.	309
SSL Client and Server Configuration on Linux, UNIX, and Windows.	309
OpenSSL in a PowerExchange SSL Environment.	310
Creating a CA Certificate with OpenSSL.	310
Creating a Personal Certificate with OpenSSL.	311
Creating a Certificate By Using the z/OS RACDCERT Command.	312
Customizing the DBMOVER Configuration File on the SSL Server.	317
Customizing the DBMOVER Configuration File on the SSL Client.	318
Verifying Connections.	319
 Chapter 8: PowerExchange Alternative Network Security.....	 320
PowerExchange Alternative Network Security Overview.	320
Modes of PowerExchange Encryption.	320
PowerExchange Encryption Standard.	321
AES Encryption Standard.	321
Setting Encryption Defaults in the DBMOVER Configuration File.	321
 Chapter 9: PowerExchange Nonrelational SQL.....	 322
PowerExchange Nonrelational SQL Overview.	322
PowerExchange Nonrelational SQL Syntax.	323
SQL Syntax for a Basic Comparison.	323
LIKE Comparison.	324
BETWEEN Comparison.	325
NULL Comparison.	325
Allowed Keywords in the SQL Syntax.	325
Accessing Data with Column Names the Same as SQL Keywords.	325
Using SQL with User-Defined Fields.	326
IMS Call Considerations.	326
DTLDESCRIBE Metadata Syntax.	326
DTLDESCRIBE TABLES Qualifier.	326

DTLDESCRIBE PROCEDURES Qualifier.	329
DTLDESCRIBE PROCEDURECOLUMNS Qualifier.	330
DTLDESCRIBE COLUMNS Qualifier.	332
DTLDESCRIBE RECORDS Qualifier.	335
DTLDESCRIBE PKEYS Qualifier.	337
DTLDESCRIBE FKEYS Qualifier.	339
DTLDESCRIBE SCHEMAS Qualifier.	341
DTLDESCRIBE Qualifiers.	342
DTLDESCRIBE Qualifiers for DB2 for i5/OS Sources.	343
DTLDESCRIBE Qualifiers for DB2 for z/OS Sources.	343
DTLDESCRIBE Qualifiers for DB2 for Linux, UNIX, and Windows Sources.	343
DTLDESCRIBE Qualifiers for NRDB Sources.	343
DTLDESCRIBE Qualifiers for NRDB2 Sources.	344
DTLDESCRIBE Qualifiers for Oracle Sources.	344
Restrictions on PowerExchange Nonrelational SQL.	344
Truncating a Target Table.	345
Inserting Rows into a Table.	345
Chapter 10: PowerExchange Globalization.	346
PowerExchange Globalization Overview.	346
Default Code Pages Installed with PowerExchange.	347
General Code Pages Available with PowerExchange.	347
Source-Specific Code Pages.	348
How PowerExchange Determines Internal Code Page Numbers by Data Source.	348
DB2 for i5/OS Code Page Parameters.	350
DB2 for z/OS Code Page Parameters.	350
Oracle Code Page Parameters.	354
User-Defined Code Pages.	355
Adding User-Defined ICU-Compatible Code Pages.	356
PowerExchange Static Code Pages.	358
ICUCHECK Utility.	360
ICUCHECK Syntax.	360
Report 1. Power Exchange Control Table for Simple Code Pages.	361
Report 2. PowerExchange Control Table for ICU Code Pages.	362
Report 3. Comparing Name and Character Size Information with ICU.	364
Report 4. Comparing ICU Code Page Information with PowerExchange.	364
Report 5. PowerExchange Code Page Names and Aliases.	365
Report 6. PM Locale Code Pages.	366
Report 7. Control Counts.	366
Report 8. Errors.	367
Code Pages Shipped with PowerExchange.	367
PowerExchange Static Code Pages.	367
Code Pages That Handle All Supported Unicode Characters.	368

Code Pages Based on ASCII Encoding.	369
ICU Code Pages Based on EBCDIC Encoding.	372
Specialized ICU Code Pages.	377
Handling Conversion Errors and Special Situations.	379
Code Page Conversion Errors.	379
Unexpected Data in a Code Page.	380
Type 1 Fallback Mappings and Multibyte Data.	381
Chapter 11: Using the PowerExchange ODBC Drivers.....	383
PowerExchange ODBC Driver Overview.	383
PowerExchange ODBC Data Sources on Windows.	383
Creating an ODBC Data Source on Windows.	384
Creating an ODBC Data Source on Windows to Be Used with the PowerExchange ODBC Unicode Driver.	385
Updating PowerExchange ODBC Data Sources on Windows.	386
PowerExchange Data Source Wizard Properties.	386
PowerExchange ODBC Data Sources on Linux or UNIX.	399
Example PowerExchange Data Source Specifications.	400
Accessing Multibyte Metadata with ODBC.	401
PowerExchange ODBC Driver Data Source Parameters.	401
SQL Escape Sequences.	414
SQL Escape Sequences with PowerCenter.	415
SQL Escape Sequences Available in PowerCenter.	415
{DTLADAPREFETCH=Y N} Escape Sequence.	416
{DTLADAPWD= } Escape Sequence.	416
{DTLAPP= } Escape Sequence.	416
{DTLARRAYSIZE= } Escape Sequence.	416
{DTLCONFWRITE=N Y T} Escape Sequence.	416
{DTLCONNOVR= } Escape Sequence.	416
{DTLDATAPWD= } Escape Sequence.	417
{DTLDB2DEGREE= } Escape Sequence.	417
{DTLDSN= } Escape Sequence.	417
{DTLEVENTTB= } Escape Sequence.	418
{DTLFILELIST=Y N} Escape Sequence.	418
{DTLIMTYPE=BA AI TU} Escape Sequence.	418
{DTLIMGOV=Y N} Escape Sequence.	418
{DTLINSMODE=LOAD UPDATE} Escape Sequence.	418
{DTLJRNL= } Escape Sequence.	418
{DTLLIBRARYLIST= } Escape Sequence.	419
{DTLLOGSID=} Escape Sequence.	419
{DTLMODELDCB=} Escape Sequence.	419
{DTLNOUPDATECDEP=Y N} Escape Sequence.	419
{DTLORACOLL= } Escape Sequence.	419

{DTLORACONN= } Escape Sequence.	419
{DTLORAINST= } Escape Sequence.	419
{DTLORASchema= } Escape Sequence.	420
{DTLOVRDBF= } Escape Sequence.	420
{DTLREJECTFILE= } Escape Sequence.	420
{DTLSESSID=} Escape Sequence.	420
{DTLSTOPONERRORS =} Escape Sequence.	420
{DTLSTRIPORDERBY=Y N} Escape Sequence.	420
{DTLTIMEOUT= } Escape Sequence.	421
{DTLUDBDB= } Escape Sequence.	421
{DTLUPDELSEINS=Y} Escape Sequence.	421
{DTLUSECATALOGMETADATA=Y N} Escape Sequence.	421
{DTLWORKERS= } Escape Sequence.	421
{DTLXTRASchema= } Escape Sequence.	421
{DTLXTYPE=RS SL} Escape Sequence.	421
Chapter 12: PowerExchange Datatypes and Conversion Matrix.	422
PowerExchange Datatype Conversions.	422
ODBC and PowerExchange Datatype Equivalencies.	423
Appendix A: DTL__CAPXTIMESTAMP Time Stamps.	425
Time Stamps That Are Reported in the DTL__CAPXTIMESTAMP Field by Data Source.	425
Appendix B: PowerExchange Glossary.	427
Index.	446

Preface

Use the *Informatica® PowerExchange® Reference Manual* to look up reference information about configuring and using PowerExchange in all types of supported environments. This publication includes information for configuring the DBMOVER configuration file, PowerExchange security, and the PowerExchange ODBC driver.

Use this publication in conjunction with the *PowerExchange Installation and Upgrade Guide*, *PowerExchange Bulk Data Movement Guide*, and PowerExchange CDC guides.

Informatica Resources

Informatica provides you with a range of product resources through the Informatica Network and other online portals. Use the resources to get the most from your Informatica products and solutions and to learn from other Informatica users and subject matter experts.

Informatica Network

The Informatica Network is the gateway to many resources, including the Informatica Knowledge Base and Informatica Global Customer Support. To enter the Informatica Network, visit <https://network.informatica.com>.

As an Informatica Network member, you have the following options:

- Search the Knowledge Base for product resources.
- View product availability information.
- Create and review your support cases.
- Find your local Informatica User Group Network and collaborate with your peers.

Informatica Knowledge Base

Use the Informatica Knowledge Base to find product resources such as how-to articles, best practices, video tutorials, and answers to frequently asked questions.

To search the Knowledge Base, visit <https://search.informatica.com>. If you have questions, comments, or ideas about the Knowledge Base, contact the Informatica Knowledge Base team at KB_Feedback@informatica.com.

Informatica Documentation

Use the Informatica Documentation Portal to explore an extensive library of documentation for current and recent product releases. To explore the Documentation Portal, visit <https://docs.informatica.com>.

If you have questions, comments, or ideas about the product documentation, contact the Informatica Documentation team at infa_documentation@informatica.com.

Informatica Product Availability Matrices

Product Availability Matrices (PAMs) indicate the versions of the operating systems, databases, and types of data sources and targets that a product release supports. You can browse the Informatica PAMs at <https://network.informatica.com/community/informatica-network/product-availability-matrices>.

Informatica Velocity

Informatica Velocity is a collection of tips and best practices developed by Informatica Professional Services and based on real-world experiences from hundreds of data management projects. Informatica Velocity represents the collective knowledge of Informatica consultants who work with organizations around the world to plan, develop, deploy, and maintain successful data management solutions.

You can find Informatica Velocity resources at <http://velocity.informatica.com>. If you have questions, comments, or ideas about Informatica Velocity, contact Informatica Professional Services at ips@informatica.com.

Informatica Marketplace

The Informatica Marketplace is a forum where you can find solutions that extend and enhance your Informatica implementations. Leverage any of the hundreds of solutions from Informatica developers and partners on the Marketplace to improve your productivity and speed up time to implementation on your projects. You can find the Informatica Marketplace at <https://marketplace.informatica.com>.

Informatica Global Customer Support

You can contact a Global Support Center by telephone or through the Informatica Network.

To find your local Informatica Global Customer Support telephone number, visit the Informatica website at the following link:

<https://www.informatica.com/services-and-training/customer-success-services/contact-us.html>.

To find online support resources on the Informatica Network, visit <https://network.informatica.com> and select the eSupport option.

CHAPTER 1

Introduction to PowerExchange

This chapter includes the following topics:

- [PowerExchange Overview, 16](#)
- [PowerExchange Components, 16](#)

PowerExchange Overview

PowerExchange can perform the following functions to synchronize a data source and data target:

- Bulk data movement to materialize or entirely refresh a data target.
- Change data capture to keep a data source and data target synchronized.

PowerExchange bulk data movement can move large amounts of data between different systems efficiently. However, frequent bulk data movement operations can be costly in terms of space, time, and staff resources.

With PowerExchange change data capture (CDC), you can capture only the changes that are made to a data source. PowerExchange CDC has minimal impact on the performance and availability of the source database, tables, and files. PowerExchange provides two methods of change data capture:

- Synchronous change data capture, which occurs in real time by integrating into the transaction performing the change
- Asynchronous change data capture, also called log-based changed data capture, which captures the changes from the source database or source relational database logs.

PowerExchange works with PowerCenter® to move bulk data and change data to a variety of data target types across an enterprise.

This manual provides reference information for customizing PowerExchange for your environment. After installing the product, review the topics on the DBMOVER configuration file parameters and product security options. Also, review any of the other topics that might pertain to your specific environment and data replication requirements.

PowerExchange Components

PowerExchange uses some or all of the following components to perform its functions, depending on the system that you are using:

PowerExchange Navigator

The graphical user interface from which you define and manage data maps, capture registrations, and extraction maps for the data sources from which you want to extract bulk data or capture change data. PowerExchange uses these definitions to determine the data sources to process. Capture registrations and extraction maps pertain to change data capture (CDC) only.

PowerExchange Listener

Manages data maps for nonrelational files and DB2 tables, and capture registrations and extraction maps for all data sources. The PowerExchange Listener maintains these definitions in the following files:

- Data maps in the DATAMAPS file
- Capture registrations in the CCT file
- Extraction maps in the DTLCAMAP or CAMAPS file

The PowerExchange Listener also handles extraction requests for bulk data and change data.

If a data source or data target is remote from the system on which you are using PowerExchange, you must also run a PowerExchange Listener on the remote data source or target to communicate with PowerExchange.

PowerExchange Agent

On a z/OS system, provides capture registration information to the following ECCRs during CDC:

- DB2
- IMS synchronous
- Batch VSAM
- CICS/VSAM

Other ECCRs read capture registration information directly from the CCT data set. For all of the ECCRs, the PowerExchange Agent verifies the capture registration information.

The PowerExchange Agent also manages global queues and data flow among various PowerExchange CDC components.

PowerExchange Logger for Linux, UNIX, and Windows

On a Linux, UNIX, or Windows machine, optionally writes change data from the change stream to PowerExchange Logger log files. PowerExchange then extracts the change data from the PowerExchange Logger log files rather than from the source database or log files. A PowerExchange Logger process can operate in either continuous mode or batch mode.

PowerExchange Logger for z/OS

On a z/OS system, receives captured change data from the PowerExchange Environmental Change Capture Routines (ECCR) that are connected to it and stores the change data in log data sets. The change data is then available for real-time extractions or PowerExchange Condense jobs.

PowerExchange Environmental Change Capture Routine (ECCR)

On a z/OS system, captures change data from a data source and passes the captured changes to the PowerExchange Logger for recording. PowerExchange provides an ECCR for each type of data source. Depending on the source type, the ECCR captures changes synchronously as they are made or asynchronously from database logs.

PowerExchange Condense

On i5/OS or z/OS, optionally creates condense files that contain a condensed version of the change data in the change stream. If you use this optional feature, PowerExchange extracts change data from condense files rather than from log files. You can run multiple Condense jobs at a time.

CHAPTER 2

DBMOVER Configuration File

This chapter includes the following topics:

- [DBMOVER Configuration File Overview, 19](#)
- [DBMOVER Statement Summary Tables, 19](#)
- [Syntax of DBMOVER Statements, 49](#)
- [DBMOVER Statements, 50](#)
- [Consumer API \(CAPI\) Connection Statements, 204](#)
- [DBMOVER Statements for Commands Issued through the pwxcmd Program, 206](#)
- [Sample DBMOVER Configuration Files, 207](#)

DBMOVER Configuration File Overview

The PowerExchange DBMOVER configuration file contains statements that control PowerExchange operations and communications. Define a DBMOVER configuration file on each PowerExchange system.

PowerExchange provides a sample DBMOVER configuration file for each supported operating system. The information in a sample file is sufficient to test the installation of the PowerExchange software. Customize the sample file. For example, you might need to configure additional PowerExchange nodes, a data source, CDC options, or localization options.

DBMOVER Statement Summary Tables

PowerExchange has many DBMOVER statements that you use to configure PowerExchange.

Some DBMOVER statements apply to specific operating systems or data sources.

The following tables list all DBMOVER statements in alphabetical order, by data source, and operating system.

Summary of All DBMOVER Statements

PowerExchange provides DBMOVER statements that you use to configure PowerExchange.

The following table briefly defines all DBMOVER statements and provides cross-references to the complete description of each statement:

Statement	Description	Reference
ABEND_SW	Controls whether PowerExchange activates additional error routines to handle abends.	“ABEND_SW Statement” on page 50
ADA_L3_ALLOW	An Adabas optimization statement that controls whether PowerExchange uses Adabas L3 commands to read records from a file in logical sequence by descriptor value.	“ADA_L3_ALLOW Statement” on page 51
ADA_TIMEZONE_CHECK	Controls whether PowerExchange issues the PWX-03206 message only once for each Adabas database that it accesses that does not have an Adabas DD statement for timezone (/TZINFO DD) defined.	“ADA_TIMEZONE_CHECK Statement” on page 52
ADABAS_DEFAULT_DBID	Specifies the DBID value that PowerExchange uses when a data map specifies 0 in the Database ID property and the bulk data movement session does not specify an override value.	“ADABAS_DEFAULT_DBID Statement” on page 52
ADABAS_MU_SEARCH	Specifies whether Adabas multi-value (MU) fields are included in search functions. An MU field is a single field that occurs a variable number of times in a record. The MU field name is prefixed with a \$ symbol and is allowed in searches. The MU field must be an Adabas descriptor field.	“ADABAS_MU_SEARCH Statement” on page 53
ADABAS_PREFIX	Specifies the prefix that PowerExchange uses to construct a user ID to access Adabas files.	“ADABAS_PREFIX Statement” on page 54
ADABASCODEPAGE	Specifies the single-byte and multibyte code pages to use for an Adabas database.	“ADABASCODEPAGE Statement” on page 54
ADAOPT	An Adabas optimization statement that controls whether PowerExchange uses Adabas L3 commands to read records from a file in logical sequence by descriptor value.	“ADAOPT Statement” on page 55
ADAOPTM	Deprecated. Exists for backward compatibility.	“ADAOPTM Statement” on page 56
ADAPREFETCH	Controls whether PowerExchange uses the Adabas prefetch feature when reading records.	“ADAPREFETCH Statement” on page 56

Statement	Description	Reference
ADASTATS	Controls whether PowerExchange writes statistical information about Adabas operations to the PowerExchange message log file.	“ADASTATS Statement” on page 56
ADAUSER	Controls whether PowerExchange uses the job name as the Adabas user ID to access Adabas files in a netport job.	“ADAUSER Statement” on page 57
APPBUFSIZE	Specifies the maximum data buffer size in bytes.	“APPBUFSIZE Statement” on page 57
APPBUFSIZEDYN	For supported data sources, specifies whether to dynamically increase the maximum data buffer size.	“APPBUFSIZEDYN Statement” on page 58
AS400EVENTMSGQ	Specifies an event message queue on an i5/OS system to which PowerExchange writes messages that do not require an operator reply.	“AS400EVENTMSGQ Statement” on page 58
AS400USRJRNCODE	Specifies two-character journal entry type codes that identify add library and delete library operations in a DB2 for i5/OS journal. Use this statement to override the default codes that are used for refreshing the CDC library interest list after adding or deleting libraries that contain instances of the same source table.	“AS400USRJRNCODE Statement” on page 59
AS400USRMSGQ	Specifies the message queue on an i5/OS system to which PowerExchange writes messages that require an operator reply.	“AS400USRMSGQ Statement” on page 60
BS	Specifies the block size that PowerExchange uses when dynamically allocating target data sets on z/OS if you do not provide a block size.	“BS Statement” on page 61
CAPI_CONN_NAME	Specifies the default CAPI_CONNECTION statement for all data source types.	“CAPI_CONN_NAME Statement” on page 61
CAPI_CONNECTION - AS4J	Specifies the Consumer API (CAPI) parameters to use for DB2 for i5/OS CDC sources.	“CAPI_CONNECTION - AS4J Statement” on page 62
CAPI_CONNECTION - CAPX	Specifies the CAPI parameters to use for continuous extraction of change data from PowerExchange Logger for Linux, UNIX, and Windows log files.	“CAPI_CONNECTION - CAPX Statement” on page 67
CAPI_CONNECTION - LRAP	Specifies the CAPI parameters to use for the Log Read API (LRAPI) component of the PowerExchange Logger for z/OS.	“CAPI_CONNECTION - LRAP Statement” on page 69
CAPI_CONNECTION - MSQL	Specifies the CAPI parameters to use for Microsoft SQL Server CDC sources.	“CAPI_CONNECTION - MSQL Statement” on page 71

Statement	Description	Reference
CAPI_CONNECTION - MYSQL	Specifies the CAPI parameters to use for MySQL CDC sources.	“CAPI_CONNECTION - MYSQL Statement” on page 77
CAPI_CONNECTION - ORAD	Specifies the CAPI parameters to use for PowerExchange Express CDC for Oracle sources.	“CAPI_CONNECTION - ORAD Statement” on page 80
CAPI_CONNECTION - PG	Specifies the CAPI parameters to use for PowerExchange PostgreSQL CDC sources.	“CAPI_CONNECTION - PG Statement” on page 82
CAPI_CONNECTION - UDB	Specifies the CAPI parameters to use for DB2 for Linux, UNIX, and Windows (LUW) CDC sources.	“CAPI_CONNECTION - UDB Statement” on page 85
CAPI_CONNECTION - UOWC	Specifies the CAPI parameters to use for the UOW Cleanser.	“CAPI_CONNECTION - UOWC Statement” on page 89
CAPI_SRC_DFLT	Specifies the default CAPI_CONNECTION statement for a specific data source type.	“CAPI_SRC_DFLT Statement” on page 94
CAPT_PATH	Specifies the path to the local directory on a Linux, UNIX, and Windows system that contains the control files for CDC.	“CAPT_PATH Statement” on page 96
CAPT_XTRA	Specifies the path to the local directory on a Linux, UNIX, and Windows system that stores extraction maps for CDC.	“CAPT_XTRA Statement” on page 97
CMDNODE	Specifies connection information for a PowerExchange process that can be the target of pwxcmd commands. Process types include the PowerExchange Listener, PowerExchange Condense, Power Exchange ECCRs on z/OS (Adabas, IDMS, or IMS log-based ECCRs, or Datacom table-based ECCR), and PowerExchange Logger for Linux, Unix and Windows.	“CMDNODE Statement” on page 97
CODEPAGE	Specifies the code pages that PowerExchange uses for operating system information, data, and literal values in SQL statements.	“CODEPAGE Statement” on page 98
COLON	Defines the character that PowerExchange uses as a the delimiter character for time values in times and time stamp fields.	“COLON Statement” on page 99
COMPRESS	Controls whether PowerExchange uses its proprietary compression routines to compress data that it transmits across the network.	“COMPRESS Statement” on page 99
CONSOLE_CODEPAGE	Specifies the code page that PowerExchange uses to display message text on consoles.	“CONSOLE_CODEPAGE Statement” on page 100

Statement	Description	Reference
CONSOLE_MSG	Specifies whether to write certain messages to the domain log in addition to writing them to the PowerExchange log file on the Integration Service machine.	"CONSOLE_MSG Statement" on page 100
CONSOLE_TRACE	Controls whether PowerExchange writes messages from TRACE statements to the DTLOUT data set for z/OS or to the QPRINT output for i5/OS, in addition to writing these messages to the PowerExchange message log file.	"CONSOLE_TRACE Statement" on page 101
CONVCHAR	Controls how PowerExchange handles conversion errors for CHAR and VARCHAR columns that PowerExchange detects during bulk data movement read operations.	"CONVCHAR Statement" on page 101
CPX_DIR	Specifies the name of the library that PowerExchange uses on i5/OS to read and store the extraction maps for DB2 for i5/OS CDC sources.	"CPX_DIR Statement" on page 102
CREDENTIALS_CASE	Controls the case that PowerExchange uses for operating system user IDs and passwords.	"CREDENTIALS_CASE Statement" on page 102
DATAMAP_SERVER	Specifies the node name of the PowerExchange Listener that acts as the server for access requests to the file that stores data maps.	"DATAMAP_SERVER Statement" on page 103
DATERANGE	Specifies the range of years that PowerExchange uses to validate data in DATE and TIMESTAMP fields if you enable data checking in a data map.	"DATERANGE Statement" on page 104
DB2_BIN_AS_CHAR	Controls whether PowerExchange considers data in CHAR and VARCHAR columns that you define with the FOR BIT DATA clause as character data.	"DB2_BIN_AS_CHAR Statement" on page 105
DB2_BIN_CODEPAGE	Defines the single-byte and multibyte CCSID values that PowerExchange uses to process character data in columns that you define with the FOR BIT DATA clause, if you specify Y for the DB2_BIN_AS_CHAR statement.	"DB2_BIN_CODEPAGE Statement" on page 105
DB2_ERRORFILE	Specifies the name of the user-customized SQL error file that PowerExchange uses for DB2 bulk data movement operations.	"DB2_ERRORFILE Statement" on page 106
DB2CODEPAGE	Defines the single-byte, mixed, and graphics CCSID values that PowerExchange uses to process bulk data from the specified DB2 subsystem.	"DB2CODEPAGE Statement" on page 106

Statement	Description	Reference
DB2DEF_ENCODING	Defines the default encoding scheme that PowerExchange assigns to any DB2 columns without an encoding scheme when you create a DB2 data map.	“DB2DEF_ENCODING Statement” on page 110
DB2ID	Defines the DB2 plan and the PowerExchange access method module that PowerExchange uses to process bulk data from the specified DB2 subsystem.	“DB2ID Statement” on page 111
DB2PLAN	Defines the DB2 plan that PowerExchange uses for bulk data movement processing.	“DB2PLAN Statement” on page 112
DECPOINT	Defines the character that PowerExchange uses as the decimal point character in fields that contain noninteger numbers.	“DECPOINT Statement” on page 112
DEFAULTCHAR	Defines the character that PowerExchange uses to replace any character data in a nonrelational data source that is not valid.	“DEFAULTCHAR Statement” on page 113
DEFAULTDATE	Defines a numeric date string that PowerExchange uses to replace any missing component in a date field.	“DEFAULTDATE Statement” on page 113
DISABLE_PARTITIONS	Controls whether PowerExchange disables partitioning of the source data for bulk data movement operations.	“DISABLE_PARTITIONS Statement” on page 113
DISP	Specifies the status, normal disposition, and abnormal disposition that PowerExchange uses when dynamically allocating target data sets on z/OS, if you do not provide DISP information.	“DISP Statement” on page 114
DM_RESOURCE	Specifies the suffix that PowerExchange uses to create the PowerExchange-specific resource profile that controls access to the DATAMAPS data set.	“DM_RESOURCE Statement” on page 115
DM_SUBTASK	Controls how the PowerExchange Listener accesses the file that stores data maps.	“DM_SUBTASK Statement” on page 115
DMXCACHE_DELETEECSA	Determines whether PowerExchange frees ECSA memory when data maps caching runs in multiple-jobs mode and no files exist with nonzero use counts in ECSA memory	“DMXCACHE_DELETEECSA Statement” on page 116
DMXCACHE_MAX_MEMORY_MB	Enables PowerExchange data maps caching and configures the maximum size for the data maps cache in megabytes.	“DMXCACHE_MAX_MEMORY_MB Statement” on page 117
DMXCACHE_MULTIPLEJOBS	Determines whether PowerExchange runs data maps caching in multiple-jobs or single-job mode.	“DMXCACHE_MULTIPLEJOBS Statement” on page 117

Statement	Description	Reference
DMX_DIR	Specifies the location that PowerExchange uses to read and store data maps.	“DMX_DIR Statement” on page 118
DTLMSG_CODEPAGE	Specifies the code page identifier that indicates which version of the PowerExchange messages file to use.	“DTLMSG_CODEPAGE Statement” on page 119
DTLMSG_LANGUAGE	Specifies the language that, along with the UTF8 code page identifier, indicates which version of the PowerExchange messages file to use.	“DTLMSG_LANGUAGE Statement” on page 119
ENCRYPT	Controls whether PowerExchange uses encryption when moving data. In addition to an internal proprietary encryption method, PowerExchange supports AES encryption. If you enter AES in the ENCRYPT statement, you must include the ENCRYPTLEVEL statement.	“ENCRYPT Statement” on page 120
ENCRYPTLEVEL	Specifies the encryption level to use if you select AES encryption. The encryption level determines the encryption key length.	“ENCRYPTLEVEL Statement” on page 121
ENQMAJORNAME	Specifies the major name for ENQ macros that PowerExchange issues to lock PowerExchange data sets and resources.	“ENQMAJORNAME Statement” on page 122
ENQSYSTEMS	Controls whether PowerExchange uses a scope of SYSTEM or SYSTEMS for ENQ macros that it issues to lock PowerExchange data sets and resources.	“ENQSYSTEMS Statement” on page 122
ERRROWNOTFOUND	Specifies whether PowerExchange generates or does not generate errors for UPDATE or DELETE operations on nonexistent rows.	“ERRROWNOTFOUND Statement” on page 123
EXT_CP_SUPPT	Controls character conversion in single-byte static code pages.	“EXT_CP_SUPPT Statement” on page 123
FILEMAPPED_MEMORY_DIR	Specifies the directory that stores temporary file-mapped memory files for PowerExchange Listener statistics on AIX PPC64, i5/OS (IBM i), and Solaris SP64 systems.	“FILEMAPPED_MEMORY_DIR Statement” on page 124
GDGLOCATE	Controls whether PowerExchange instructs dynamic allocation to use a LOCATE to get the most recent catalog information for generation data set allocation requests.	“GDGLOCATE Statement” on page 125
GSBUFSIZE	Enables you to increase the maximum internal buffer size for PowerExchange processing of a row of data when the default buffer size is not adequate.	“GSBUFSIZE Statement” on page 126

Statement	Description	Reference
HOSTNAME	Specifies an i5/OS host name that overrides the host name that PowerExchange retrieves from the TCP/IP Host table for communications with the PowerExchange Listener on i5/OS.	“HOSTNAME Statement” on page 126
ICUALIAS	Adds or deletes an alias that points to a custom ICU code page.	“ICUALIAS Statement” on page 127
ICUCNVPROPERTY	Sets customized properties for an ICU code page.	“ICUCNVPROPERTY Statement” on page 127
ICUCONVERTER	Adds a code page converter to or deletes a code page converter from an ICU code page.	“ICUCONVERTER Statement” on page 130
ICUDATADIR	Defines the directory location for the binary CNV files that makeconv produces.	“ICUDATADIR Statement” on page 131
IMSBSDS	Identifies the high-level qualifier of the IMS bootstrap data set, which is used when reading source metadata from the IMS catalog. Also indicates the order in which PowerExchange searches for DBD information in the DBDLIB library or IMS catalog.	“IMSBSDS Statement” on page 132
IMSID	Defines the IMS subsystem information that PowerExchange uses for IMS bulk data movement operations and for IMS log-based CDC.	“IMSID Statement” on page 133
JOBCLASS	Specifies the job class that PowerExchange uses for CA IDMS/DB metadata and DB2 LOAD utility batch jobs.	“JOBCLASS Statement” on page 134
LDAP_BASE	Specifies the location within an LDAP directory from which to conduct a PowerExchange user search.	“LDAP_BASE Statement” on page 134
LDAP_BIND_DN	Specifies the distinguished name (DN) of an LDAP user with sufficient access rights to connect to LDAP and conduct a search for the PowerExchange user.	“LDAP_BIND_DN Statement” on page 134
LDAP_BIND_EPWD	Specifies the encrypted password for the LDAP search user.	“LDAP_BIND_EPWD Statement” on page 135
LDAP_BIND_PWD	Specifies the password for the LDAP search user.	“LDAP_BIND_PWD Statement” on page 136
LDAP_BIND_TIMEOUT	Specifies the number of seconds that a PowerExchange Listener or PowerExchange Logger for Linux, UNIX, and Windows waits to authenticate a search user or PowerExchange user against LDAP.	“LDAP_BIND_TIMEOUT Statement” on page 136

Statement	Description	Reference
LDAP_FILTER	Specifies a filter to restrict the LDAP search for a PowerExchange user.	“LDAP_FILTER Statement” on page 137
LDAP_HOST	Defines TCP/IP host details for any LDAP servers that PowerExchange might use for user validation.	“LDAP_HOST Statement” on page 137
LDAP_LOGIN_ATTRIBUTE	Specifies an LDAP attribute to use as a search key to match against a PowerExchange user ID.	“LDAP_LOGIN_ATTRIBUTE Statement” on page 138
LDAP_OPENSSL	Specifies Transport Layer Security (TLS) certificate information for a TLS connection to an LDAP server when PowerExchange is configured to use the OpenLDAP client.	“LDAP_OPENSSL Statement” on page 138
LDAP_PORT	Defines the TCP/IP port for LDAP servers that PowerExchange might use for user validation.	“LDAP_PORT Statement” on page 139
LDAP_SASL_MECH	Specifies the authentication mechanism to use for the Simple Authentication and Security Layer (SASL) that is used with LDAP certificate-based security.	“LDAP_SASL_MECH Statement” on page 140
LDAP_SCOPE	Defines how deep to search in the LDAP tree within the search base in LDAP searches.	“LDAP_SCOPE Statement” on page 140
LDAP_SEARCH_TIMEOUT	Specifies the number of seconds that a PowerExchange Listener waits to receive a search result during PowerExchange user validation against LDAP.	“LDAP_SEARCH_TIMEOUT Statement” on page 141
LDAP_TLS	Specifies Transport Layer Security (TLS) certificate information for a TLS connection to an LDAP server when PowerExchange is configured to use the Oracle LDAP client.	“LDAP_TLS Statement” on page 141
LISTENER	Defines the TCP/IP port on which a named PowerExchange Listener process listens for work requests.	“LISTENER Statement” on page 142
LOADCTLFILE	Specifies the PDS data set that contains the control card template member for DB2 for z/OS LOAD utility batch jobs.	“LOADCTLFILE Statement” on page 144
LOADJOBFILE	Specifies the PDS data set that contains the JCL template member for DB2 for z/OS LOAD utility and CA IDMS/DB metadata retrieval batch jobs.	“LOADJOBFILE Statement” on page 145
LOG_CODEPAGE	Specifies the code page that PowerExchange uses to write messages to the log file.	“LOG_CODEPAGE Statement” on page 145

Statement	Description	Reference
LOG_LINE_LIMIT	Specifies the maximum line length for PowerExchange messages in the message log.	“LOG_LINE_LIMIT Statement” on page 146
LOGPATH	Defines the directory location where PowerExchange writes message log files on Linux, UNIX, and Windows systems.	“LOGPATH Statement” on page 146
LOGSID	Specifies the location of the CA IDMS/DB logs and the PowerExchange log catalog.	“LOGSID Statement” on page 147
LOWVALUES	Specifies whether PowerExchange preserves hexadecimal '0' values in a VSAM or sequential data source when writing to a VSAM or sequential target.	“LOWVALUES Statement” on page 147
LRECL	Specifies the logical record length that PowerExchange uses when dynamically allocating target data sets on z/OS, if you do not provide LRECL information.	“LRECL Statement” on page 148
LU00FILE	Specifies the partitioned data set extended (PDSE) that you allocated to hold database descriptor (DBD) output from the IMS Catalog Builder Utility, DFS3LU00.	“LU00FILE Statement” on page 148
MAXTASKS	Defines the maximum number of tasks that can run concurrently in a PowerExchange Listener.	“MAXTASKS Statement” on page 149
MSGPREFIX	Defines the string that PowerExchange uses to prefix PowerExchange messages.	“MSGPREFIX Statement” on page 150
MSGPREFIX-HYPHEN	Specifies whether PowerExchange includes or excludes the hyphen character between the message prefix and the message number.	“MSGPREFIX-HYPHEN Statement” on page 150
MSS_ERRORFILE	Specifies the name of the user-customized SQL error file that PowerExchange uses for Microsoft SQL Server bulk data movement operations.	“MSS_ERRORFILE Statement” on page 151
MSSQL_SERVER_CONNECT_TIMEOUT	Specifies the timeout interval, in seconds, for a PowerExchange Navigator, DTLUCBRG utility, or DTLURDMO utility connection to a Microsoft SQL Server Management Objects (SMO) interface to manage SQL Server publications.	“MSSQL_SERVER_CONNECT_TIMEOUT Statement” on page 149
MSSQL_SERVER_STATEMENT_TIMEOUT	Specifies the timeout interval, in seconds, for processing a Transact-SQL statement issued by the PowerExchange Navigator, DTLUCBRG utility, or DTLURDMO utility to the Microsoft SQL Server Management Objects (SMO) interface.	“MSSQL_SERVER_STATEMENT_TIMEOUT Statement” on page 150

Statement	Description	Reference
MVSDB2AF	Specifies which DB2 for z/OS attachment facility PowerExchange uses for DB2 bulk data movement operations.	“MVSDB2AF Statement” on page 151
NEGSIGN	Defines the character that PowerExchange uses as the negative sign character in fields that contain negative numbers.	“NEGSIGN Statement” on page 152
NETPORT	Defines information about netport jobs and associates the netport job with a specific listener port.	“NETPORT Statement” on page 152
NODE	Defines the TCP/IP host name and port that PowerExchange uses to contact a PowerExchange Listener process.	“NODE Statement” on page 155
NOGETHOSTBYNAME	Controls whether PowerExchange gets the IP address of the local system to verify the license and include it in PowerExchange messages.	“NOGETHOSTBYNAME Statement” on page 157
NRDB_WRITE_CHAR_NULL_FILL	Defines the character or hexadecimal value that PowerExchange uses to replace null characters in a field.	“NRDB_WRITE_CHAR_NULL_FILL Statement” on page 157
NRDB_WRITE_NUM_NULL_FILL	Defines the numeric or hexadecimal value that PowerExchange uses to replace null values in an unpacked numeric field.	“NRDB_WRITE_NUM_NULL_FILL Statement” on page 157
NUMERICSIGN	Controls whether PowerExchange delivers unsigned fields as a positive X'C' or as unsigned X'F'.	“NUMERICSIGN Statement” on page 158
ODBASUPP	Controls whether PowerExchange can use the Open Database Access (ODBA) interface to access IMS databases and unload data sets for bulk data movement operations.	“ODBASUPP Statement” on page 158
ORA_ERRORFILE	Specifies the name of the user-customized SQL error file that PowerExchange uses for Oracle bulk data movement operations.	“ORA_ERRORFILE Statement” on page 159
ORACLE_CAPTURE_TYPE	Specifies PowerExchange Express CDC for Oracle as the CDC solution to use for the PowerExchange installation.	“ORACLE_CAPTURE_TYPE Statement” on page 159
ORACLE_UNHANDLED_NUMASCHAR	Controls whether PowerExchange handles the following types of numeric columns in Oracle source tables in a manner that prevents data loss: <ul style="list-style-type: none"> - TLSV1. The peer supports TLS version 1. - NUMBER columns that have a precision greater than 28 or an undefined length - FLOAT columns that have a precision greater than 15 significant digits 	“ORACLE_UNHANDLED_NUMASCHAR Statement” on page 160

Statement	Description	Reference
ORACLECODEPAGE	Specifies the PowerExchange and PowerCenter code pages to use for a specific Oracle database used in Oracle bulk data movement, if the NLS_LANG environment variable specifies a character set other than UTF8 or AL32UTF8.	"ORACLECODEPAGE Statement" on page 161
ORACLEID	Specifies the Oracle source database and connection information for PowerExchange Express CDC for Oracle.	"ORACLEID Statement" on page 162
OUSP	Loads the z/OS OpenEdition security environment when using ADABAS. Use at the direction of Informatica Global Customer Support.	"OUSP Statement" on page 164
PC_AUTH	Controls whether the PowerExchange Listener uses its MVS Program Call (PC) services routine to acquire the authorization to access CA IDMS/DB.	"PC_AUTH Statement" on page 164
PIPE	Defines the character that PowerExchange uses to separate fields in a TXT output file.	"PIPE Statement" on page 165
POLLTIME	Defines the amount of time, in thousandths of a second, that the PowerExchange Listener waits before polling for outstanding connections.	"POLLTIME Statement" on page 165
PRE861_COMPAT	Controls whether PowerExchange automatically attempts to use offload processing in PowerExchange 8.6.1 and later.	"PRE861_COMPAT Statement" on page 166
PRGIND	Controls whether PowerExchange writes or does not write read progress messages to the PowerExchange message log file.	"PRGIND Statement" on page 166
PRGINT	Defines the interval after which PowerExchange writes read progress messages to the PowerExchange message log file, if requested.	"PRGINT Statement" on page 167
PWXSOMAXCONN	Specifies the maximum number of TCP/IP socket connections that a PowerExchange Listener uses to listen for work.	"PWXSOMAXCONN Statement" on page 167
RACF_CLASS	Specifies the RACF class name that PowerExchange uses when checking PowerExchange-specific resource profiles.	"RACF_CLASS Statement" on page 168
RDBMSINSRTDFLT	Controls whether PowerExchange uses DB2 default values for DB2 columns that you define with the WITH DEFAULT clause.	"RDBMSINSRTDFLT Statement" on page 168

Statement	Description	Reference
RECFM	Specifies the record format that PowerExchange uses when dynamically allocating target data sets on z/OS, if you do not provide RECFM information.	"RECFM Statement" on page 169
REJECT_FILE_DELIMITER	Defines the character that PowerExchange uses as the default delimiter character in a reject file.	"REJECT_FILE_DELIMITER Statement" on page 169
RELEASE	Controls whether PowerExchange requests or does not request to release unused space for data sets that it dynamically allocates.	"RELEASE Statement" on page 170
RMTRDBDIRE	Defines the name of the DB2 for i5/OS database on the local system that contains the source tables for CDC in a remote journaling environment.	"RMTRDBDIRE Statement" on page 170
RMTSYSNAME	Defines the name of the i5/OS host system that contains the DB2 source tables for CDC, local journals, and journal receivers.	"RMTSYSNAME Statement" on page 170
SECURITY	Controls whether PowerExchange completes user authentication, and controls access to resources and commands.	"SECURITY Statement" on page 171
SECURITY_MSGSUPP	Controls whether to suppress the messages that are issued by RACF or a similar z/OS security product, such as CA Top Secret or ACF2, when verifying the logon credentials of a PowerExchange job or started task.	"SECURITY_MSGSUPP Statement" on page 175
SERVICE_TIMEOUT	Specifies the time, in seconds, that a PowerExchange Listener or Logger waits to receive heartbeat data from the associated Listener Service or Logger Service before shutting down and issuing an error message.	"SERVICE_TIMEOUT Statement" on page 175
SESSID	Specifies the value to use as the DB2 Correlation ID for DB2 requests.	"SESSID Statement" on page 176
SHOW_THREAD_PERF	Specifies the number of records PowerExchange processes before writing statistics messages about multithreaded processing to the PowerExchange message log file.	"SHOW_THREAD_PERF Statement" on page 176
SPACE	Specifies the units and the primary and secondary space values that PowerExchange uses when dynamically allocating target data sets on z/OS, if you do not provide SPACE information.	"SPACE Statement" on page 177
SSL	Specifies SSL certificate information for a Secure Sockets Layer (SSL) connection.	"SSL Statement" on page 177

Statement	Description	Reference
SSL_ALLOW_SELFSIGNED	Specifies the type of certificate to use for SSL security. You can use either self-signed certificates or certificates from a commercial certificate authority that the system trusts.	“SSL_ALLOW_SELFSIGNED Statement” on page 178
SSL_CIPHER_LIST	Restricts the available ciphers that a client offers to a server during an SSL handshake to the specified list.	“SSL_CIPHER_LIST Statement” on page 179
SSL_CONTEXT_METHOD	Selects the TLS version that the peer supports for PowerExchange SSL communication.	“SSL_CONTEXT_METHOD Statement” on page 180
SSL_REQ_CLNT_CERT	Controls whether a PowerExchange server authenticates the identity of a PowerExchange client by using an SSL certificate.	“SSL_REQ_CLNT_CERT Statement” on page 180
SSL_REQ_SRVR_CERT	Controls whether a PowerExchange client authenticates the identity of a PowerExchange server by using an SSL certificate.	“SSL_REQ_SRVR_CERT Statement” on page 181
START_UP_USER_EXIT	Specifies the name and programming language of a user exit program that PowerExchange calls each time the PowerExchange Listener starts or shuts down. Use this statement to enable the PowerExchange Listener to decrypt an Adabas database that is encrypted with an Adabas cipher code.	“START_UP_USER_EXIT Statement” on page 181
STATS	Controls whether PowerExchange writes SMF statistics records for the PowerExchange Listener to SMF or to a file.	“STATS Statement” on page 182
SUBMITTIMEOUT	Specifies the time, in seconds, that a PowerExchange Listener waits to receive notification from a spawned batch job that it has started.	“SUBMITTIMEOUT Statement” on page 184
SUP_FUNC	Defines the function name of the zIIP enclave for the PowerExchange Listener to the IBM Workload Manager.	“SUP_FUNC Statement” on page 184
SUP_REQUEST	Defines the name of the requestor for zIIP facilities.	“SUP_REQUEST Statement” on page 185
SUP_SSNAME	Defines the subsystem name that identifies the PowerExchange Listener started task to the IBM Workload Manager when zIIP exploitation is enabled.	“SUP_SSNAME Statement” on page 185

Statement	Description	Reference
SUP_SSTYPE	Defines the name that the IBM Workload Manager uses as the subsystem type for the enclave for zIIP.	“SUP_SSTYPE Statement” on page 185
SUPPRESS_DATA_LOGGING	Controls whether PowerExchange writes a message that displays the first 40 characters of any row that fails the data checking options specified in a data map.	“SUPPRESS_DATA_LOGGING Statement” on page 186
SVCNODE	Specifies the TCP/IP port on which a PowerExchange process listens for pwxcmd or infacmd commands. Process types include the PowerExchange Listener, PowerExchange Condense, Power Exchange ECCRs on z/OS (Adabas, IDMS, or IMS log-based ECCRs, or Datacom table-based ECCR), and PowerExchange Logger for Linux, Unix and Windows.	“SVCNODE Statement” on page 186
SYSOUT_TIMESTAMP	Controls whether PowerExchange includes time stamp information in output files and console output.	“SYSOUT_TIMESTAMP Statement” on page 188
TAPEWAIT	Controls whether netport jobs wait or do not wait for tape volumes.	“TAPEWAIT Statement” on page 188
TCPIP_ACCEPT_NONBLOCK	Specifies whether TCP/IP ports on which the PowerExchange Listener listens are blocking or nonblocking.	“TCPIP_ACCEPT_NONBLOCK Statement” on page 188
TCPIP_DIAGNOSTICS_TRACE	Controls the issuing of network diagnostic messages PWX-33316, PWX-33317, and PWX-33318.	“TCPIP_DIAGNOSTICS_TRACE Statement” on page 189
TCPIP_DIAGNOSTICS_TRACE_SZ	Controls the number of trace records to keep for network send and receive operations.	“TCPIP_DIAGNOSTICS_TRACE_SZ Statement” on page 189
TCPIPBUFFSIZE	Deprecated. Exists for backward compatibility.	“TCPIPBUFFSIZE Statement” on page 191
TCPIPVER	Specifies alternative TCP/IP socket code that PowerExchange uses instead of the standard IBM TCP/IP UNIX socket code.	“TCPIPVER Statement” on page 191
TEMPHLQ	Overrides the high-level qualifier that PowerExchange uses by default when creating a temporary file for CA IDMS/DB metadata.	“TEMPHLQ Statement” on page 191
TEXT_EOF_FOR_BINARY	Controls whether PowerExchange ignores hexadecimal value 1A when reading binary flat files on Linux, UNIX, or Windows.	“TEXT_EOF_FOR_BINARY Statement” on page 192

Statement	Description	Reference
TIMEZONE	Specifies a local time zone that PowerExchange uses for reading or writing Adabas date-time values for bulk data movement sessions.	“TIMEZONE Statement” on page 192
TRACE	Activates traces in PowerExchange code paths for diagnostic purposes.	“TRACE Statement” on page 193
TRACING	Activates PowerExchange alternative logging and specifies attributes for the alternative log files.	“TRACING Statement” on page 193
UNIT	Specifies the generic or esoteric unit name that PowerExchange uses when dynamically allocating target data sets on z/OS, if you do not provide UNIT information.	“UNIT Statement” on page 199
USE_DB_AUTH	When LDAP authentication is enabled, controls whether or not LDAP validation applies to relational connections.	“USE_DB_AUTH Statement” on page 199
USE_TYPE1_FALLBACKS	Enables type 1 fallback mappings for all code pages.	“USE_TYPE1_FALLBACKS Statement” on page 200
USESUP	Controls whether PowerExchange uses zIIP for zIIP-enabled PowerExchange Listener functions.	“USESUP Statement” on page 200
VOLSER	Specifies the volume serial number that PowerExchange uses when dynamically allocating target data sets on z/OS, if you do not provide VOLSER information.	“VOLSER Statement” on page 201
VSAM	Specifies the number of buffers that PowerExchange uses for data and index control intervals when processing VSAM data sets.	“VSAM Statement” on page 201
WAITDSN	Controls whether netport jobs wait for in-use data sets.	“WAITDSN Statement” on page 202
WORKCLASS	Defines the transaction name for Workload Manager classification when zIPP exploitation is enabled.	“WORKCLASS Statement” on page 202
WRT_ERROR_HANDLING	Indicates whether to use an alternative method of handling error messages returned from the PowerExchange Listener for bulk data movement sessions that use writer partitioning.	“WRT_ERROR_HANDLING Statement” on page 203

Summary of Data Source-Specific DBMOVER Statements

Some DBMOVER statements apply only to a specific data source type.

Adabas DBMOVER Statements

You can include DBMOVER statements specific to Adabas data sources.

The following table briefly describes the DBMOVER statements specific to Adabas data sources and provides cross-references to the complete description of the statements:

Statement	Description	Reference
ADA_L3_ALLOW	An Adabas optimization statement that controls whether PowerExchange uses Adabas L3 commands to read records from a file in logical sequence by descriptor value.	“ADA_L3_ALLOW Statement” on page 51
ADA_TIMEZONE_CHECK	Controls whether PowerExchange issues the PWX-03206 message only once for each Adabas database that it accesses that does not have an Adabas DD statement for timezone (/TZINFO DD) defined.	“ADA_TIMEZONE_CHECK Statement” on page 52
ADABAS_DEFAULT_DBID	Specifies the DBID value that PowerExchange uses when a data map specifies 0 in the Database ID property and the bulk data movement session does not specify an override value.	“ADABAS_DEFAULT_DBID Statement” on page 52
ADABAS_MU_SEARCH	Specifies whether Adabas multi-value (MU) fields are included in search functions. An MU field is a single field that occurs a variable number of times in a record. The MU field name is prefixed with a \$ symbol and is allowed in searches. The MU field must be an Adabas descriptor field.	“ADABAS_MU_SEARCH Statement” on page 53
ADABAS_PREFIX	Specifies the prefix that PowerExchange uses to construct a user ID to access Adabas files.	“ADABAS_PREFIX Statement” on page 54
ADABASCODEPAGE	Specifies the single-byte and multibyte code pages to use for an Adabas database.	“ADABASCODEPAGE Statement” on page 54
ADAOPT	An Adabas optimization statement that controls whether PowerExchange uses Adabas L3 commands to read records from a file in logical sequence by descriptor value.	“ADAOPT Statement” on page 55
ADAOPTM	Deprecated. Exists for backward compatibility.	“ADAOPTM Statement” on page 56
ADAPREFETCH	Controls whether PowerExchange uses the Adabas prefetch feature when reading records.	“ADAPREFETCH Statement” on page 56
ADASTATS	Controls whether PowerExchange writes statistical information about Adabas operations to the PowerExchange message log file.	“ADASTATS Statement” on page 56
ADAUSER	Controls whether PowerExchange uses the job name as the Adabas user ID to access Adabas files in a netport job.	“ADAUSER Statement” on page 57
OUSP	Loads the z/OS OpenEdition security environment when using ADABAS. Use at the direction of Informatica Global Customer Support.	“OUSP Statement” on page 164

Statement	Description	Reference
START_UP_USER_EXIT	Specifies the name and programming language of a user exit program that PowerExchange calls each time the PowerExchange Listener starts or shuts down. Use this statement to enable the PowerExchange Listener to decrypt an Adabas database that is encrypted with an Adabas cipher code.	“START_UP_USER_EXIT Statement” on page 181
TIMEZONE	Specifies a local time zone that PowerExchange uses to read or write Adabas date-time values for bulk data movement sessions. This statement applies to Adabas 8.2.2 data sources or targets that include date-time fields for which the TZ option is defined.	“TIMEZONE Statement” on page 192

CA IDMS/DB DBMOVER Statements

PowerExchange provides DBMOVER statements specific to CA IDMS/DB data sources.

The following table briefly defines the DBMOVER statements specific to CA IDMS/DB data sources and provides cross-references to the complete description of the statements:

Statement	Description	Reference
LOADJOBFILE	Specifies the PDS data set that contains the JCL template member for DB2 for z/OS LOAD utility and CA IDMS/DB metadata retrieval batch jobs.	“LOADJOBFILE Statement” on page 145
LOGSID	Specifies the location of the CA IDMS/DB logs and the PowerExchange log catalog.	“LOGSID Statement” on page 147
PC_AUTH	Controls whether the PowerExchange Listener uses its MVS Program Call (PC) services routine to acquire the authorization to access CA IDMS/DB.	“PC_AUTH Statement” on page 164
TEMPHLQ	Overrides the high-level qualifier that PowerExchange uses by default when creating a temporary file for CA IDMS/DB metadata.	“TEMPHLQ Statement” on page 191

DB2 for i5/OS DBMOVER Statements

You can include DBMOVER statements specific to DB2 for i5/OS data sources.

The following table briefly defines the DBMOVER statements specific to DB2 for i5/OS data sources and provides cross-references to the complete description of the statements:

Statement	Description	Reference
AS400EVENTMSGQ	Specifies an event message queue on an i5/OS system to which PowerExchange writes messages that do not require an operator reply.	"AS400EVENTMSGQ Statement" on page 58
AS400USRJRNCODE	Specifies two-character journal entry type codes that identify add library and delete library operations in a DB2 for i5/OS journal. Use this statement to override the default codes that are used for refreshing the CDC library interest list after adding or deleting libraries that contain instances of the same source table.	"AS400USRJRNCODE Statement" on page 59
AS400USRMSGQ	Specifies the message queue on an i5/OS system to which PowerExchange writes messages that require an operator reply.	"AS400USRMSGQ Statement" on page 60
CAPL_CONNECTION - AS4J	Specifies the Consumer API (CAPI) parameters needed for DB2 for i5/OS CDC sources.	"CAPL_CONNECTION - AS4J Statement" on page 62
DB2_BIN_AS_CHAR	Controls whether PowerExchange considers data in CHAR and VARCHAR columns that you define with the FOR BIT DATA clause as character data.	"DB2_BIN_AS_CHAR Statement" on page 105
DB2_BIN_CODEPAGE	Defines the single-byte and multibyte CCSID values that PowerExchange uses to process character data in columns that you define with the FOR BIT DATA clause, if you specify Y for the DB2_BIN_AS_CHAR statement.	"DB2_BIN_CODEPAGE Statement" on page 105
DB2_ERRORFILE	Specifies the name of the user-customized SQL error file that PowerExchange uses for DB2 bulk data movement operations.	"DB2_ERRORFILE Statement" on page 106
RMTRDBDIRE	Defines the name of the DB2 for i5/OS database on the local system that contains the source tables for CDC in a remote journaling environment.	"RMTRDBDIRE Statement" on page 170
RMTSYSNAME	Defines the name of the i5/OS host system that contains the DB2 source tables for CDC, local journals, and journal receivers.	"RMTSYSNAME Statement" on page 170

DB2 for Linux, UNIX, and Windows DBMOVER Statements

You can include DBMOVER statements specific to DB2 for Linux, UNIX, and Windows data sources.

The following table briefly describes the DBMOVER statements specific to DB2 for Linux, UNIX, and Windows data sources and provides cross-references to the complete description of the statements:

Statement	Description	Reference
CAPL_CONNECTION - UDB	Specifies the CAPI parameters for DB2 for Linux, UNIX, and Windows CDC sources.	“CAPL_CONNECTION - UDB Statement” on page 85
DB2_ERRORFILE	Specifies the name of the user-customized SQL error file that PowerExchange uses for DB2 bulk data movement operations.	“DB2_ERRORFILE Statement” on page 106

DB2 for z/OS DBMOVER Statements

PowerExchange provides DBMOVER statements specific to DB2 for z/OS data sources.

The following table briefly defines the DBMOVER statements specific to DB2 for z/OS data sources and provides cross-references to the complete description of the statements:

Statement	Description	Reference
DB2_ERRORFILE	Specifies the name of the user-customized SQL error file that PowerExchange uses for DB2 bulk data movement operations.	“DB2_ERRORFILE Statement” on page 106
DB2CODEPAGE	Defines the single-byte, mixed, and graphics CCSID values that PowerExchange uses to process bulk data from the specified DB2 subsystem.	“DB2CODEPAGE Statement” on page 106
DB2DEF_ENCODING	Defines the default encoding scheme that PowerExchange assigns to any DB2 columns without an encoding scheme when you create a DB2 data map.	“DB2DEF_ENCODING Statement” on page 110
DB2ID	Defines the DB2 plan and the PowerExchange access method module that PowerExchange uses to process bulk data from the specified DB2 subsystem.	“DB2ID Statement” on page 111
DB2PLAN	Defines the DB2 plan that PowerExchange uses for bulk data movement processing.	“DB2PLAN Statement” on page 112
LOADCTLFILE	Specifies the PDS data set that contains the control card template member for DB2 for z/OS LOAD utility batch jobs.	“LOADCTLFILE Statement” on page 144
LOADJOBFILE	Specifies the PDS data set that contains the JCL template member for DB2 for z/OS LOAD utility and CA IDMS/DB metadata retrieval batch jobs.	“LOADJOBFILE Statement” on page 145

Statement	Description	Reference
MVSD2AF	Specifies which DB2 for z/OS attachment facility PowerExchange uses for DB2 bulk data movement operations.	“MVSD2AF Statement” on page 151
SESSID	Specifies the value to use as the DB2 Correlation ID for DB2 requests.	“SESSID Statement” on page 176

IMS DBMOVER Statements

PowerExchange provides DBMOVER statements specific to IMS data sources.

The following table briefly defines the DBMOVER statements specific to IMS data sources and provides cross-references to the complete description of the statements:

Statement	Description	Reference
IMSBSDS	Identifies the high-level qualifier of the IMS bootstrap data set, which is used when reading source metadata from the IMS catalog. Also indicates the order in which PowerExchange searches for DBD information in the DBDLIB library or IMS catalog.	“IMSBSDS Statement” on page 132
IMSID	Defines the IMS subsystem information that PowerExchange uses for IMS bulk data movement operations and for IMS log-based CDC.	“IMSID Statement” on page 133
LU00FILE	Specifies the partitioned data set extended (PDSE) that you allocated to hold database descriptor (DBD) output from the IMS Catalog Builder Utility, DFS3LU00.	“LU00FILE Statement” on page 148
ODBASUPP	Controls whether PowerExchange can use the Open Database Access (ODBA) interface to access IMS databases and unload data sets for bulk data movement operations.	“ODBASUPP Statement” on page 158

Microsoft SQL Server DBMOVER Statements

PowerExchange provides DBMOVER statements that are specific to Microsoft SQL Server data sources.

The following table briefly defines the DBMOVER statements specific to Microsoft SQL Server data sources and provides cross-references to the complete description of the statements:

Statement	Description	Reference
CAPL_CONNECTION - MSQ	Specifies the CAPL parameters for Microsoft SQL Server CDC sources.	“CAPL_CONNECTION - MSQ Statement” on page 71
MSS_ERRORFILE	Specifies the name of the user-customized SQL error file that PowerExchange uses for Microsoft SQL Server bulk data movement operations.	“MSS_ERRORFILE Statement” on page 151

Statement	Description	Reference
MSSQL_SERVER_CONNECT_TIMEOUT	Specifies the timeout interval, in seconds, for a PowerExchange Navigator, DTLUCBRG utility, or DTLURDMO utility connection to a Microsoft SQL Server Management Objects (SMO) interface to manage SQL Server publications.	"MSSQL_SERVER_CONNECT_TIMEOUT Statement" on page 149
MSSQL_SERVER_STATEMENT_TIMEOUT	Specifies the timeout interval, in seconds, for processing a Transact-SQL statement issued by the PowerExchange Navigator, DTLUCBRG utility, or DTLURDMO utility to the Microsoft SQL Server Management Objects (SMO) interface.	"MSSQL_SERVER_STATEMENT_TIMEOUT Statement" on page 150

MySQL DBMOVER Statements

PowerExchange provides DBMOVER statements that are specific to MySQL data sources.

The following table briefly defines the only DBMOVER statement that is specific to MySQL data sources and provides a cross-reference to the complete description of the statement:

Statement	Description	Reference
CAPI_CONNECTION - MYSQL	Specifies the CAPI parameters for MySQL CDC sources.	"CAPI_CONNECTION - MYSQL Statement" on page 77

Nonrelational DBMOVER Statements

You can include DBMOVER statements specific to nonrelational data sources.

The following table briefly defines the DBMOVER statements specific to nonrelational data sources and provides cross-references to the complete description of the statements:

Statement	Description	Reference
DEFAULTCHAR	Defines the character that PowerExchange uses to replace any character data in a nonrelational data source that is not valid.	"DEFAULTCHAR Statement" on page 113
NRDB_WRITE_CHAR_NULL_FILL	Defines the character or hexadecimal value that PowerExchange uses to replace null characters in a field.	"NRDB_WRITE_CHAR_NULL_FILL Statement" on page 157

Statement	Description	Reference
NRDB_WRITE_NUM_NULL_FILL	Defines the numeric or hexadecimal value that PowerExchange uses to replace null values in an unpacked numeric field.	“NRDB_WRITE_NUM_NULL_FILL Statement” on page 157
TEXT_EOF_FOR_BINARY	Controls whether PowerExchange ignores hexadecimal value 1A when reading binary flat files on Linux, UNIX, or Windows.	“TEXT_EOF_FOR_BINARY Statement” on page 192

Oracle DBMOVER Statements

You can include DBMOVER statements specific to Oracle data sources.

The following table briefly defines the DBMOVER statements specific to Oracle data sources and provides cross-references to the complete description of the statements:

Statement	Description	Reference
CAPICONNECTION - ORAD	Specifies the CAPI parameters to use for PowerExchange Express CDC for Oracle sources.	“CAPICONNECTION - ORAD Statement” on page 80
ORA_ERRORFILE	Specifies the name of the user-customized SQL error file that PowerExchange uses for Oracle bulk data movement operations.	“ORA_ERRORFILE Statement” on page 159
ORACLE_CAPTURE_TYPE	Specifies that PowerExchange Express CDC for Oracle is the CDC solution to use for a PowerExchange installation.	“ORACLE_CAPTURE_TYPE Statement” on page 159
ORACLE_UNHANDLED_NUMASCHAR	Controls whether PowerExchange handles the following types of numeric columns in Oracle source tables in a manner that prevents data loss: <ul style="list-style-type: none"> - NUMBER columns that have a precision greater than 28 or an undefined length - FLOAT columns that have a precision greater than 15 significant digits 	“ORACLE_UNHANDLED_NUMASCHAR Statement” on page 160

Statement	Description	Reference
ORACLECODEPAGE	Specifies the PowerExchange and PowerCenter code pages to use for a specific Oracle database used in Oracle bulk data movement, if the NLS_LANG environment variable specifies a character set other than UTF8 or AL32UTF8.	“ORACLECODEPAGE Statement” on page 161
ORACLEID	Specifies the Oracle source database and connection information for PowerExchange Express CDC for Oracle.	“ORACLEID Statement” on page 162

PostgreSQL DBMOVER Statements

PowerExchange provides DBMOVER statements that are specific to PostgreSQL data sources.

The following table briefly defines the only DBMOVER statement that is specific to PostgreSQL CDC data sources and provides a cross-reference to the complete description of the statement:

Statement	Description	Reference
CAPI_CONNECTION - PG	Specifies the CAPI parameters for PostgreSQL CDC sources.	“CAPI_CONNECTION - PG Statement” on page 82

VSAM DBMOVER Statements

PowerExchange provides DBMOVER statements specific to VSAM data sources.

The following table briefly defines the DBMOVER statements specific to VSAM data sources and provides cross-references to the complete description of the statements:

Statement	Description	Reference
VSAM	Specifies the number of buffers that PowerExchange uses for data and index control intervals when processing VSAM data sets.	“VSAM Statement” on page 201

Summary of Operating System-Specific DBMOVER Statements

Some DBMOVER statements apply only to a specific operating system.

i5/OS DBMOVER Statements

PowerExchange provides DBMOVER statements specific to the i5/OS operating system.

The following table briefly defines the DBMOVER statements specific to the i5/OS operating system and provides cross-references to the complete description of the statements:

Statement	DescriptionDescription	Reference
AS400EVENTMSGQ	Specifies an event message queue on an i5/OS system to which PowerExchange writes messages that do not require an operator reply.	"AS400EVENTMSGQ Statement" on page 58
AS400USRJRNCODE	Specifies two-character journal entry type codes that identify add library and delete library operations in a DB2 for i5/OS journal. Use this statement to override the default codes that are used for refreshing the CDC library interest list after adding or deleting libraries that contain instances of the same source table.	"AS400USRJRNCODE Statement" on page 59
AS400USRMSGQ	Specifies the message queue on an i5/OS system to which PowerExchange writes messages that require an operator reply.	"AS400USRMSGQ Statement" on page 60
CONSOLE_TRACE	Controls whether PowerExchange writes messages from TRACE statements to the MVS console for z/OS or to the QPRINT output for i5/OS, in addition to writing these messages to the PowerExchange message log file.	"CONSOLE_TRACE Statement" on page 101
CPX_DIR	Specifies the name of the library that PowerExchange uses on i5/OS to read and store the extraction maps for DB2 for i5/OS CDC sources.	"CPX_DIR Statement" on page 102
FILEMAPPED_MEMORY_DIR	Specifies the directory that stores temporary file-mapped memory files for PowerExchange Listener statistics on AIX PPC64, i5/OS (IBM i), and Solaris SP64 systems.	"FILEMAPPED_MEMORY_DIR Statement" on page 124
HOSTNAME	Specifies an i5/OS host name that overrides the host name that PowerExchange retrieves from the TCP/IP Host table for communications with the PowerExchange Listener on i5/OS.	"HOSTNAME Statement" on page 126
RMTRDBDIRE	Defines the name of the DB2 for i5/OS database on the local system that contains the source tables for CDC in a remote journaling environment.	"RMTRDBDIRE Statement" on page 170

Linux, UNIX, and Windows DBMOVER Statements

PowerExchange provides DBMOVER statements specific to the Linux, UNIX, and Windows operating systems.

The following table briefly defines the DBMOVER statements specific to the Linux, UNIX, and Windows operating systems and provides cross-references to the complete description of the statements:

Statement	Description	Reference
CAPT_PATH	Specifies the path to the local directory on a Linux, UNIX, and Windows system that contains the control files for CDC.	"CAPT_PATH Statement" on page 96
CAPT_XTRA	Specifies the path to the local directory on a Linux, UNIX, and Windows system that stores extraction maps for CDC.	"CAPT_XTRA Statement" on page 97
CMDNODE	Linux, UNIX, and Windows. Specifies connection information for a PowerExchange process to which you want to send pwxcmd commands.	"CMDNODE Statement" on page 97
CONSOLE_CODEPAGE	Linux, UNIX, and Windows. Specifies the code page that PowerExchange uses to display message text on consoles.	"CONSOLE_CODEPAGE Statement" on page 100
CONSOLE_MSG	Specifies whether to write certain messages to the domain log in addition to writing them to the PowerExchange log file on the Integration Service machine.	"CONSOLE_MSG Statement" on page 100
FILEMAPPED_MEMORY_DIR	Specifies the directory that stores temporary file-mapped memory files for PowerExchange Listener statistics on AIX PPC64, i5/OS (IBM i), and Solaris SP64 systems.	"FILEMAPPED_MEMORY_DIR Statement" on page 124
LDAP_BASE	Specifies the location within an LDAP directory from which to conduct a PowerExchange user search.	"LDAP_BASE Statement" on page 134
LDAP_BIND_DN	Specifies the distinguished name (DN) of an LDAP user with sufficient access rights to connect to LDAP and conduct a search for the PowerExchange user.	"LDAP_BIND_DN Statement" on page 134
LDAP_BIND_EPWD	Specifies the encrypted password for the LDAP search user.	"LDAP_BIND_EPWD Statement" on page 135
LDAP_BIND_PWD	Specifies the password for the LDAP search user.	"LDAP_BIND_PWD Statement" on page 136
LDAP_BIND_TIMEOUT	Specifies the number of seconds that a PowerExchange Listener or Logger waits to authenticate a user against LDAP.	"LDAP_BIND_TIMEOUT Statement" on page 136
LDAP_FILTER	Specifies a filter to speed up or restrict the LDAP search for a PowerExchange user.	"LDAP_FILTER Statement" on page 137

Statement	Description	Reference
LDAP_HOST	Defines TCP/IP host details for any LDAP servers that PowerExchange might use for user validation.	“LDAP_HOST Statement” on page 137
LDAP_LOGIN_ATTRIBUTE	Specifies an LDAP attribute to use as a search key to match against a PowerExchange user ID.	“LDAP_LOGIN_ATTRIBUTE Statement” on page 138
LDAP_OPENSSL	Specifies Transport Layer Security (TLS) certificate information for a TLS connection to an LDAP server when PowerExchange is configured to use the OpenLDAP client.	“LDAP_OPENSSL Statement” on page 138
LDAP_PORT	Defines the TCP/IP port for LDAP servers that PowerExchange might use for user validation.	“LDAP_PORT Statement” on page 139
LDAP_SASL_MECH	Specifies the authentication mechanism to be used for Simple Authentication and Security Layer (SASL) that is used with LDAP certificate-based security.	“LDAP_SASL_MECH Statement” on page 140
LDAP_SCOPE	Defines how deep to search in the LDAP tree within the search base in LDAP searches.	“LDAP_SCOPE Statement” on page 140
LDAP_SEARCH_TIMEOUT	Specifies the time, in seconds, that a PowerExchange Listener waits to receive a search result during PowerExchange user validation against LDAP.	“LDAP_SEARCH_TIMEOUT Statement” on page 141
LDAP_TLS	Specifies Transport Layer Security (TLS) certificate information for a TLS connection to an LDAP server when PowerExchange is configured to use the Oracle LDAP client.	“LDAP_TLS Statement” on page 141
LOG_CODEPAGE	Windows. Specifies the code page that PowerExchange uses to write messages to the log file.	“LOG_CODEPAGE Statement” on page 145
LOGPATH	Linux, UNIX, and Windows. Defines the directory location where PowerExchange writes message log files on Linux, UNIX, and Windows systems.	“LOGPATH Statement” on page 146
TEXT_EOF_FOR_BINARY	Linux, UNIX, and Windows. Controls whether PowerExchange ignores hexadecimal value 1A when reading binary flat files on Linux, UNIX, or Windows.	“TEXT_EOF_FOR_BINARY Statement” on page 192
USE_DB_AUTH	When LDAP authentication is enabled, controls whether or not LDAP validation applies to relational connections.	“USE_DB_AUTH Statement” on page 199

z/OS DBMOVER Statements

PowerExchange provides DBMOVER statements specific to the z/OS operating system.

The following table briefly defines the DBMOVER statements specific to the z/OS operating system and provides cross-references to the complete description of the statements:

Statement	Description	Reference
ABEND_SW	Controls whether PowerExchange activates additional error routines to handle abends.	"ABEND_SW Statement" on page 50
BS	Specifies the block size that PowerExchange uses when dynamically allocating target data sets on z/OS if you do not provide a block size.	"BS Statement" on page 61
CAPL_CONNECTION - LRAP	Specifies the CAPL parameters needed for the Log Read API (LRAP) component of the PowerExchange Logger for z/OS.	"CAPL_CONNECTION - LRAP Statement" on page 69
CONSOLE_TRACE	Controls whether PowerExchange writes messages from TRACE statements to the MVS console for z/OS or to the QPRINT output for i5/OS, in addition to writing these messages to the PowerExchange message log file.	"CONSOLE_TRACE Statement" on page 101
DATAMAP_SERVER	Specifies the node name of the PowerExchange Listener that acts as the server for access requests to the file that stores data maps.	"DATAMAP_SERVER Statement" on page 103
DISP	Specifies the status, normal disposition, and abnormal disposition that PowerExchange uses when dynamically allocating new target data sets on z/OS, if no DISP is provided.	"DISP Statement" on page 114
DM_RESOURCE	Specifies the suffix that PowerExchange uses to create the PowerExchange-specific resource profile that controls access to the DATAMAPS data set.	"DM_RESOURCE Statement" on page 115
DM_SUBTASK	Controls how the PowerExchange Listener accesses the file that stores data maps.	"DM_SUBTASK Statement" on page 115
DMXCACHE_DELETEECSA	Determines whether PowerExchange frees ECSA memory when data maps caching runs in multiple-jobs mode and no files exist with nonzero use counts in ECSA memory.	"DMXCACHE_DELETEECSA Statement" on page 116
DMXCACHE_MAX_MEMORY_MB	Enables PowerExchange data maps caching and configures the maximum size for the data maps cache in megabytes.	"DMXCACHE_MAX_MEMORY_MB Statement" on page 117
DMXCACHE_MULTIPLEJOBS	Determines whether PowerExchange runs data maps caching in multiple-jobs or single-job mode.	"DMXCACHE_MULTIPLEJOBS Statement" on page 117

Statement	Description	Reference
ENQMAJORNAME	Specifies the major name for ENQ macros that PowerExchange issues to lock PowerExchange data sets and resources.	"ENQMAJORNAME Statement" on page 122
ENQSYSTEMS	Controls whether PowerExchange uses a scope of SYSTEM or SYSTEMS for ENQ macros that it issues to lock PowerExchange data sets and resources.	"ENQSYSTEMS Statement" on page 122
GDGLOCATE	Controls whether PowerExchange instructs dynamic allocation to use a LOCATE to get the most recent catalog information for generation data set allocation requests.	"GDGLOCATE Statement" on page 125
JOBCLASS	Specifies the job class that PowerExchange uses for CA IDMS/DB metadata and DB2 LOAD utility batch jobs.	"JOBCLASS Statement" on page 134
LRECL	Specifies the logical record length that PowerExchange uses when dynamically allocating target data sets on z/OS, if you do not provide LRECL information.	"LRECL Statement" on page 148
MVSDDB2AF	Specifies which DB2 for z/OS attachment facility PowerExchange uses for DB2 bulk data movement operations.	"MVSDDB2AF Statement" on page 151
NETPORT	Defines information about netport jobs and associates the netport job with a specific listener port.	"NETPORT Statement" on page 152
PC_AUTH	Controls whether the PowerExchange Listener uses its MVS Program Call (PC) services routine to acquire the authorization to access DBMS load libraries.	"PC_AUTH Statement" on page 164
RACF_CLASS	Specifies the RACF class name that PowerExchange uses when checking PowerExchange-specific resource profiles.	"RACF_CLASS Statement" on page 168
RECFM	Specifies the record format that PowerExchange uses when dynamically allocating target data sets on z/OS, if you do not provide RECFM information.	"RECFM Statement" on page 169
RELEASE	Controls whether PowerExchange requests or does not request to release unused space for data sets that it dynamically allocates.	"RELEASE Statement" on page 170
SPACE	Specifies the units and the primary and secondary space values that PowerExchange uses when dynamically allocating target data sets on z/OS, if you do not provide SPACE information.	"SPACE Statement" on page 177

Statement	Description	Reference
START_UP_USER_EXIT	Specifies the name and programming language of a user exit program that PowerExchange calls each time the PowerExchange Listener starts or shuts down. Use this statement to enable the PowerExchange Listener to decrypt an Adabas database that is encrypted with an Adabas cipher code.	"START_UP_USER_EXIT Statement" on page 181
STATS	Controls whether PowerExchange writes SMF statistics records for the PowerExchange Listener to SMF or to a file.	"STATS Statement" on page 182
SUBMITTIMEOUT	Specifies the time, in seconds, that a PowerExchange Listener waits to receive notification from a spawned batch job that it has started.	"SUBMITTIMEOUT Statement" on page 184
SUP_FUNC	Defines the function name of the zIIP enclave for the PowerExchange Listener to the IBM Workload Manager.	"SUP_FUNC Statement" on page 184
SUP_REQUEST	Defines the name of the requestor for zIIP facilities.	"SUP_REQUEST Statement" on page 185
SUP_SSNAME	Defines the subsystem name that identifies the PowerExchange Listener started task to the IBM Workload Manager when zIIP exploitation is enabled.	"SUP_SSNAME Statement" on page 185
SUP_SSTYPE	Defines the name that the IBM Workload Manager uses as the subsystem type for the enclave for zIIP.	"SUP_SSTYPE Statement" on page 185
TAPEWAIT	Controls whether netport jobs wait or do not wait for tape volumes.	"TAPEWAIT Statement" on page 188
TCPIP_ACCEPT_NONBLOCK	Specifies whether TCP/IP ports on which the PowerExchange Listener listens are blocking or nonblocking.	"TCPIP_ACCEPT_NONBLOCK Statement" on page 188
UNIT	Specifies the generic or esoteric unit name that PowerExchange uses when dynamically allocating target data sets on z/OS, if you do not provide UNIT information.	"UNIT Statement" on page 199
USESUP	Controls whether PowerExchange uses zIIP for zIIP-enabled PowerExchange Listener functions.	"USESUP Statement" on page 200
VOLSER	Specifies the volume serial number that PowerExchange uses when dynamically allocating target data sets on z/OS, if you do not provide VOLSER information.	"VOLSER Statement" on page 201

Statement	Description	Reference
WAITDSN	Controls whether netport jobs wait for in-use data sets.	"WAITDSN Statement" on page 202
WORKCLASS	Defines the default transaction name for Workload Manager classification when zIIP exploitation is enabled.	"WORKCLASS Statement" on page 202

Syntax of DBMOVER Statements

When you define DBMOVER statements and parameters, follow syntax rules.

The syntax for a statement shows the following information:

- Required and optional parameters and options
- Defaults for parameters
- How to define the statement, parameters, and options

This topic uses specific notational conventions to describe the syntax of the DBMOVER statements.

Syntax Rules and Guidelines for DBMOVER Statements

You must follow syntax rules and guidelines when you include statements in the DBMOVER configuration file.

Use the following rules and guidelines for statements in the DBMOVER configuration file:

- Start every statement on a new line.
- If a statement or parameter has more than one option, include them in parentheses and separate them by a comma (,) or a space. For example:

```
NODE=(node_name,TCPIP,host_name,2480)
```

- You can span statements across lines but specify only one parameter on each line. For example:

```
DB2CODEPAGE=(DSN9,
              EBCDIC CCSID=(037,037,037),
              MIXED=Y)
```

- If you need to include a space in a value such as a Windows path, enclose the complete parameter value in double quotation marks. For example:

```
"aaa bbbb\cccc"
```

Note: Use straight quotes ("). Do not use curly quotes (").

- Begin comment statements with /*. PowerExchange does not parse comment statements. For example:

```
/* this is a comment
```

You can include comment statements between DBMOVER statements but not between parameters of a single, multiline DBMOVER statement.

Notational Conventions for DBMOVER Statements

The syntax descriptions of the DBMOVER statements use notational conventions.

The following table describes the notational conventions used in the DBMOVER statement syntax and provides examples:

Notation	Description
<i>italic</i>	Italic indicates variables that you must replace with a name or value. For example: <code>ADABAS_DEFAULT_DBID=<i>dbid</i></code>
{ }	Single braces enclose alternative entries. Use only one of the entries. Do not type the braces when you enter the option. For example: <code>ABEND_SW={<u>N</u> Y}</code>
[]	Brackets surround optional parameters for a statement. Do not type the brackets when you enter the parameter. For example: <code>DB2ID=(<i>db2_subsystem</i>[,<i>plan</i>][,<i>module_name</i>])</code>
	A vertical bar indicates a mutually exclusive choice. When used with brackets, enter one or none of the items. When used with braces, you must enter one of the items. For example: <code>CREDENTIALS_CASE={A <u>D</u> S}</code>
<u> </u>	Underline indicates the default used if you do not specify any of the alternatives. For example: <code>ABEND_SW={<u>N</u> Y}</code>
,	A comma separates the parameters on a statement. For example: <code>DB2ID=(<i>db2_subsystem</i>[,<i>plan</i>][,<i>module_name</i>])</code>
...	An ellipsis indicates additional parameters that do not appear in the syntax. For example: <code>TYPE=(AS4J, ...)</code>

DBMOVER Statements

When you configure the DBMOVER configuration file, you define statements and parameters.

A statement can be required or optional.

ABEND_SW Statement

The ABEND_SW statement controls whether PowerExchange activates additional error routines to handle abends.

Operating Systems: z/OS

Required: No

Syntax:

```
ABEND_SW={N|Y}
```

Valid Values:

- **N.** PowerExchange uses the standard z/OS Language Environment (LE) error handling routines and allows LE to handle all signals.
- **Y.** PowerExchange activates additional error routines to handle abends that clean up acquired storage and enqueues for the following C program signals:
 - SIGABND
 - SIGABRT
 - SIGFPE
 - SIGILL
 - SIGSEGV

Default is N.

ADA_L3_ALLOW Statement

The ADA_L3_ALLOW statement is an Adabas optimization statement that controls whether PowerExchange uses Adabas L3 commands to read records from a file in logical sequence by descriptor value.

Operating Systems: z/OS

Data Sources: Adabas

Related Statements: ADAOPT and ADASTATS

Required: No

Syntax:

```
ADA_L3_ALLOW={N|Y}
```

Valid Values:

- **N.** PowerExchange attempts to determine if the Adabas version that is running supports beginning and ending ranges. If PowerExchange determines that the Adabas version supports ranges, PowerExchange uses L3 commands. Otherwise, PowerExchange uses L2 commands.
- **Y.** If you specify Adabas descriptors on the WHERE clause of an SQL SELECT statement, PowerExchange uses L3 commands to read records in logical order from the Adabas file. If the SQL statement does not contain a WHERE clause, PowerExchange uses L2 commands to read records from the file in the physical order in which Adabas stored them.

PowerExchange does not verify that the Adabas version in use supports beginning and ending descriptor key ranges with L3 commands.

Specify Y if PowerExchange cannot correctly determine the running Adabas version.

Default is N.

Usage Notes:

- To use L3 commands, you must also specify Y for the ADAOPT statement.

- Unlike the ADAOPT statement, the ADA_L3_ALLOW statement does not cause PowerExchange to verify that Adabas version 7 or later is running. Use the ADA_L3_ALLOW statement when user modifications to Adabas prevent PowerExchange from verifying which Adabas version is installed.
- By default, PowerExchange selects **Ranges Only** in the **Optimization Level** list in data maps. If you select **OFF** in the **Optimization Level** list, PowerExchange ignores the ADA_L3_ALLOW statement and does not use L3 commands when processing data for that data map.
- If you specify Y for the ADASTATS statement, PowerExchange writes message PWX-02196 in the PowerExchange message log file. This message indicates whether you can use L3 commands and the status of optimization.

ADA_TIMEZONE_CHECK Statement

The ADA_TIMEZONE_CHECK statement controls whether PowerExchange issues the PWX-03206 message only once for each Adabas database that it accesses that does not have an Adabas DD statement for timezone (//TZINFO DD) defined. PWX-03206 indicates that the Adabas DD statement for timezone was not found in the Adabas nucleus JCL.

Operating Systems: z/OS

Data Sources: Adabas

Related Statements: TIMEZONE

Required: No

Syntax:

```
ADA_TIMEZONE_CHECK={N|Y}
```

Valid Values:

- **N.** PowerExchange issues PWX-03206 only once for each Adabas database that it tries to access that does not have an Adabas DD statement for timezone defined.
- **Y.** PowerExchange issues a PWX-03206 message for each access to an Adabas database that does not have an Adabas DD statement for timezone defined.

Default is Y.

Usage Notes:

- If the issuance of multiple PWX-03206 messages might cause problems, such as when PowerExchange accesses an Adabas database to perform a Lookup transformation, specify ADA_TIMEZONE_CHECK=N.

ADABAS_DEFAULT_DBID Statement

The ADABAS_DEFAULT_DBID statement specifies the DBID value that PowerExchange uses when a data map specifies 0 in the **Database ID** property and the bulk data movement session does not specify an override value.

Operating Systems: z/OS

Data Sources: Adabas

Required: No

Syntax:

```
ADABAS_DEFAULT_DBID={dbid|0}
```

Value: For the *dbid* variable, enter a number from 0 through 65535. Default is 0.

Usage Notes:

- You can include leading zeros in the DBID value. For example, to specify 100 as the default DBID, you can define any of the following statements:
 - ADABAS_DEFAULT_DBID=100
 - ADABAS_DEFAULT_DBID=0100
 - ADABAS_DEFAULT_DBID=00100
- To ease migration of data maps from one PowerExchange environment to another, enter 0 in the **Database ID** property of the data map. Then, define an ADABAS_DEFAULT_DBID statement in each environment to provide the DBID value. Alternatively, to override the DBID value specified in a data map, set a value in the **Database Id Override** attribute of the bulk data movement session in PowerCenter or, if you use ODBC, in the ODBC parameters.

ADABAS_MU_SEARCH Statement

The ADABAS_MU_SEARCH statement specifies whether Adabas multi-value (MU) fields are included in search functions. An MU field is a single field that occurs a variable number of times in a record. The MU field name is prefixed with a \$ symbol and is allowed in searches. The MU field must be an Adabas descriptor field. It appears as type MU in an Adabas record description.

Operating Systems: z/OS

Data Sources: Adabas

Required: No

Syntax:

```
ADABAS_MU_SEARCH={Y|N}
```

Valid Values:

- **Y.** PowerExchange uses multi-value Adabas descriptor fields in searches. If you specify a multi-value field in a WHERE clause of an SQL SELECT statement, PowerExchange scans the field for values specified in the search.
- **N.** PowerExchange will not include Adabas multi-value fields in searches.

Default is N.

Usage Notes:

- In an existing data map definition, the field name for a multi-value descriptor (DE) field can be used by adding the \$ prefix to the multi-value field name and eliminating the subscripting index.
- Limit search criteria that includes multi-value fields to an EQUAL comparator or a BETWEEN phrase. Complex searches using an OR comparator or non-descriptor search values are not supported. The following SELECT statements show simple search criteria for multi-value fields:

```
SELECT * from table_name where $MU_field = 'JONES'
SELECT * from table_name where $MU_field >= 10 and $MU_field <= 20
SELECT * from table_name where $MU_field is between 10 and 20
SELECT AA_field, AB_field, $MU_field where $MU_field = 10
```

- The multi-value field can also be used in a SQL SELECT statement if the field was used in the search criteria. The SQL SELECT statement presents the contents of the multi-value array data that meets the search criteria.
- If you set the ADABAS_MU_SEARCH statement to Y, make sure that the ADAOPT statement is also set to Y to use L3 commands to read records from file in logical sequence by descriptor value. If the search cannot be optimized, and there are OR criteria in the SQL, the search will fail.

ADABAS_PREFIX Statement

The ADABAS_PREFIX statement specifies the prefix that PowerExchange uses to construct a user ID to access Adabas files.

Operating Systems: z/OS

Data Sources: Adabas

Related Statements: ADAUSER, for netport jobs

Required: No

Syntax:

```
ADABAS_PREFIX={prefix|DTL0}
```

Value: For the *prefix* variable, enter a one- to four-character alphanumeric prefix. To construct the user ID, PowerExchange appends an additional four characters to generate a unique value for each Adabas file that the PowerExchange Listener opens. Default is DTL0.

Usage Notes:

- To access the same Adabas file simultaneously with different user IDs, specify a unique user ID. If each user ID that PowerExchange constructs to access an Adabas file is not unique, one or more PowerExchange Listeners might be unable to access the Adabas file. In this case, the read operation fails with Adabas Response Code 48 Sub Code 8 and PowerExchange message PWX-00416.
- To ensure a unique Adabas user ID when multiple PowerExchange Listeners access the same Adabas file, use the ADABAS_PREFIX statement to specify a different prefix for each PowerExchange Listener. If the user ID is not unique, the first PowerExchange Listener that tries to access the Adabas file succeeds, and the second PowerExchange Listener fails. Also, the PowerCenter session fails.
- If you use netport jobs to access Adabas files, define the ADAUSER statement to ensure a unique user ID for each netport job.

ADABASCODEPAGE Statement

The ADABASCODEPAGE statement specifies the single-byte and multibyte code pages to use for an Adabas database.

Enter up to 20 ADABASCODEPAGE statements in the DBMOVER configuration file.

Operating Systems: z/OS

Data Sources: Adabas

Related Statements: CODEPAGE

Required: No

Syntax:

```
ADABASCODEPAGE=(dbid  
                  [, single_cp]  
                  [, multi_cp])
```

Parameters:

dbid

Required. The Adabas database identifier (DBID) to which the code page or code pages apply.

Specify 0 to indicate the Adabas default database.

single_cp

Optional. The name of a single-byte code page.

multi_cp

Optional. The name of a multibyte code page. If the database contains WIDECHAR fields, enter a multibyte code page.

Usage Notes:

- Usually, you specify this statement for databases that have WIDECHAR fields that require a code page other than the default code page or the code page that is set in the CODEPAGE statement.

When you define a data map, you can override the code pages that this statement specifies. In the data map, specify a code page for a specific source file, a wide-character code page for all WIDECHAR fields, or specific code pages for each field. A field-level code page overrides a data map code page, and a data map code page overrides any code page that this statement or the CODEPAGE statement specify.

ADAOPT Statement

The ADAOPT statement is an Adabas optimization statement that controls whether PowerExchange uses Adabas L3 commands to read records from a file in logical sequence by descriptor value.

Operating Systems: z/OS

Data Sources: Adabas

Related Statements: ADA_L3_ALLOW, and ADASTATS

Required: No

Syntax:

ADAOPT={N|Y}

Valid Values:

- **N.** PowerExchange disables optimization and uses L2 commands to read records from files.
- **Y.** If you specify Adabas descriptors on the WHERE clause of a SQL SELECT statement, PowerExchange uses L3 commands to read records in logical order from the Adabas file. If the SQL statement does not contain a WHERE clause, PowerExchange uses L2 commands to read records from the file in the physical order in which they were stored by Adabas.

PowerExchange verifies that the Adabas version that is being used supports beginning and ending descriptor key ranges with L3 commands.

If you have installed user modifications to Adabas, PowerExchange might be unable to correctly determine the installed version of Adabas. In this case, you must also specify Y for the ADA_L3_ALLOW statement to use L3 commands.

Default is Y.

Usage Notes:

- By default, PowerExchange selects **Ranges Only** in the **Optimization Level** list in data maps. If you select **OFF** in the **Optimization Level** list, PowerExchange ignores this statement and does not use L3 commands when processing data for that data map.
- If you specify Y for the ADASTATS statement, PowerExchange writes message PWX-02196 in the PowerExchange message log file. This message indicates whether you can enter L3 commands and the status of optimization.

ADAOPTM Statement

The ADAOPTM statement is deprecated and exists for backward compatibility only.

The ADAOPTM statement previously controlled how PowerExchange used descriptors on Adabas L3 commands. The option that you select in the **Optimization Level** list in the data map now overrides the value specified for this statement.

ADAPREFETCH Statement

The ADAPREFETCH statement controls whether PowerExchange uses the Adabas prefetch feature to improve performance when reading records.

Operating Systems: z/OS

Data Sources: Adabas

Required: No

Syntax:

```
ADAPREFETCH={N|Y}
```

Valid Values:

- **N.** PowerExchange does not use the Adabas prefetch feature when reading records.
- **Y.** PowerExchange uses the Adabas prefetch feature when reading records to improve read performance.

Default is N.

Caution: On UNIX and Windows, Adabas does not allow prefetch processing with the ACBX interface. If you are using ACBX on one of these systems, and ADAPREFETCH=Y, PowerExchange reverts to ACB calls, which cannot read records larger than 32 KB. If you want to use ACBX on UNIX or Windows, do not specify the ADAPREFETCH=Y statement.

ADASTATS Statement

The ADASTATS statement controls whether PowerExchange writes statistical information about Adabas operations to the PowerExchange message log file.

Operating Systems: z/OS

Data Sources: Adabas

Related Statements: ADA_L3_ALLOW, and ADAOPT

Required: No

Syntax:

```
ADASTATS={N|Y}
```

Valid Values:

- **N.** PowerExchange does not write Adabas statistics messages to the PowerExchange message log file.
- **Y.** PowerExchange writes Adabas statistics messages to the PowerExchange message log file.

Before reading records, PowerExchange writes messages that contain the following Adabas information:

- Whether PowerExchange uses prefetch
- Whether PowerExchange uses L2 or L3 commands to read data

If PowerExchange uses L3 commands, PowerExchange writes additional messages that contain the key and key values used with L3 commands.

- The format buffer that PowerExchange uses

After reading records, PowerExchange issues a message that displays the number of L2 and L3 commands used to read the data.

Default is N.

ADAUUSER Statement

The ADAUSER statement controls whether PowerExchange uses the job name as the Adabas user ID to access Adabas files in a netport job.

Operating Systems: z/OS

Data Sources: Adabas netport jobs

Related Statements: ADABAS_PREFIX, for PowerExchange Listener access

Required: No

Syntax:

```
ADAUSER={DEFAULT|JOBNAME|userid}
```

Valid Values:

- **DEFAULT.** PowerExchange uses either the default value of DTL00011 for the Adabas user ID or, if specified, the ADABAS_PREFIX value appended with 0011.
- **JOBNAME.** PowerExchange uses the job name of the netport job for the Adabas user ID.
Specify JOBNAME if you run multiple netport jobs that access the same Adabas file. Otherwise, if multiple jobs run under the same user ID, such as the default, the jobs might fail with Adabas Response Code 48 Sub Code 8 and PowerExchange message PWX-00416.
- **userid.** PowerExchange uses the specified value without change as the Adabas user ID.

Default is DEFAULT.

APPBUFSIZE Statement

The APPBUFSIZE statement specifies the maximum buffer size, in bytes, for reading or writing data.

When the buffer size is reached, PowerExchange sends the buffer data across the network to the system that requests the data and then starts using another buffer.

If dynamic application buffer sizing is enabled, the APPBUFSIZE statement specifies the initial maximum buffer size. Dynamic application buffer sizing is enabled by default. You can explicitly enable it by specifying Y for the APPBUFSIZEDYN statement in the DBMOVER configuration file.

Operating Systems: All

Related Statements: APPBUFSIZEDYN

Required: No

Syntax:

```
APPBUFSIZE={buffer_size|256000}
```

Value: For the *buffer_size* variable, enter a value that is greater than the maximum size of a single row. Valid values are from 34816 through 8388608. Default is 256000.

Usage Notes:

- For bulk data movement sessions that use reader or writer partitions, you can increase the APPBUFSIZE value to help improve session performance.

APPBUFSIZEDYN Statement

The APPBUFSIZEDYN statement specifies whether to enable dynamic application buffer sizing.

The DBMOVER APPBUFSIZE statement defines the initial size of the application buffer for all connections made during a PowerExchange Listener run. If APPBUFSIZEDYN=Y, PowerExchange resizes the application buffers for individual connection as needed.

The APPBUFSIZEDYN statement applies to PowerExchange connections to data sources with either fixed-length or variable-length records. A variable-length record is a record with at least one variable-length field. A variable-length field has a datatype of VARCHAR or VARBIN.

For each connection to a data source with variable-length records, PowerExchange resizes the application buffer when it encounters a record that is too large to fit into the buffer. PowerExchange increases the size of the application buffer to a value of ten times the size of the record that has overflowed, up to the maximum application buffer size of 8 MB. The new size remains in effect for the duration of the Listener run or until the application buffer is resized again. PowerExchange never decreases the application buffer size for a connection after the Listener run has started.

For each connection to a data source with fixed-length records, PowerExchange determines the record length when the connection is opened and resizes the application buffer once, up to the maximum application buffer size of 8 MB, as needed.

Operating Systems: All

Data Sources: All

Related Statements: APPBUFSIZE

Required: No

Syntax:

```
APPBUFSIZEDYN={N|Y}
```

Valid Values:

- **N.** PowerExchange does not perform dynamic application buffer sizing.
- **Y.** PowerExchange performs dynamic application buffer sizing.

Default is Y.

AS400EVENTMSGQ Statement

The AS400EVENTMSGQ statement specifies the message queue on an i5/OS system to which PowerExchange writes messages that do *not* require a user reply.

Operating Systems: i5/OS

Data Sources: DB2 for i5/OS

Related Statements: CAPI_CONNECTION - AS4J, AS400USRMSGQ

Required: No

Syntax:

```
AS400EVENTMSGQ=library/queue_name
```

Parameters: Separate the following parameters by a slash (/):

library

The name of the library that contains the message queue.

queue_name

The message queue name.

Usage Notes: PowerExchange writes messages to the specified message queue when ALWCLRPFM=Y, ALWLIBRFSH=Y, or AS4JRNEXIT=Y is specified on the AS4J CAPI_CONNECTION statement, as follows:

- If ALWCLRPFM=Y, PowerExchange writes the DTL3002 message to the message queue if PowerExchange encounters a journal entry for a CLRPFM command. PowerExchange cannot replicate changes that a CLRPFM command makes.
- If ALWLIBRFSH=Y, PowerExchange writes message DTL3003I for an *ADDLIB (add library) event and message DTL3004I for a *DTLLIB (delete library) event to the message queue when you issue the SNDPWXJRNE command with an **Event Identifier** value. This command triggers refresh processing of the CDC library interest list for the added or deleted libraries.
- If AS4JRNEXIT=Y, PowerExchange writes the DTL3001 message for each journal receiver that a change data extraction processes. PowerExchange writes the message when it reads the first journal entry for the next journal receiver on the chain. If multiple change data extractions successfully process the same journal receiver, the message queue contains multiple DTL3001 messages for that journal receiver.

For more information about this message queue, see the *PowerExchange CDC Guide for i5/OS*.

AS400USRJRNCODE Statement

The AS400USRJRNCODE defines two-character journal entry type codes that identify SNDPWXJRNE add library and delete library entries in a DB2 for i5/OS user journal. Use this statement to override the default journal entry type codes that the SNDPWXJRNE command uses.

The SNDPWXJRNE command issues requests for refreshing the CDC library interest list for added or deleted libraries. The libraries must contain instances of the same source table, and the table instances must be journaled to the same user journal.

This statement applies to all of the DB2 for i5/OS instances that are defined in AS4J CAPI_CONNECTION statements with the ALWLIBRFSH=Y setting in the same DBMOVER member.

Operating Systems: i5/OS

Data Sources: DB2 for i5/OS

Related Statements: CAPI_CONNECTION - AS4J

Required: No

Syntax:

```
AS400USRJRNCODE=(ADDLIB=jrn_entry_type_code,DTLLIB=jrn_entry_type_code)
```

Use a comma (,) to separate the ADDLIB and DTLLIB parameters.

Parameters:

ADDLIB=journal_entry_type_code

Optional. Enter a unique two-character journal entry type code, such as U1, that the SNDPWXJRNE command writes to the journal for an SNDPWXJRNE add library (*ADDLIB) entry. Overrides the default code of I1.

DLTLIB=journal_entry_type_code

Optional. Enter a unique two-character journal entry type code, such as U2, that the SNDPWXJRNE command writes to the journal for a SNDPWXJRNE delete library (*DLTLIB) entry. Overrides the default code of I2.

Usage Notes:

- For PowerExchange to use this statement, you must set the ALWLIBRSH parameter in the AS4J CAPI_CONNECTION statement to Y. This setting enables you to use the SNDPWXJRNE command to refresh the CDC library interest list after you add or delete i5/OS libraries that contain instances of the same DB2 source table and that are journaled to the same user journal, without restarting real-time PowerCenter PWXPC CDC sessions.
- By default, the SNDPWXJRNE command uses I1 as the entry type code for an add library request and uses I2 for a delete library request. If these journal entry type codes are used by another application, you must specify unique entry type codes with this AS400USRJRNCODE statement.
- The journal entry codes that you specify in this statement, or the default journal entry codes, are not specific to the journals that are identified in the AS4J CAPI_CONNECTION statements in the same DBMOVER member. PowerCenter CDC sessions can override these journals at runtime.

For more information about event message queues, see the *PowerExchange CDC Guide for i5/OS*.

AS400USRMSGQ Statement

The AS400USRMSGQ statement specifies the message queue on an i5/OS system to which PowerExchange writes messages that require a user reply. These messages typically require a user reply of Y or N, or Yes or No.

Operating Systems: i5/OS

Data Sources: DB2 for i5/OS

Related Statements: CAPI_CONNECTION - AS4J, AS400EVENTMSGQ

Required: No

Syntax:

```
AS400USRMSGQ=library/queue_name
```

Parameters: Separate the following parameters by a forward slash (/):

library

The name of the library that contains the message queue for user reply messages.

queue_name

The message queue name.

If you do not specify this statement, PowerExchange writes messages that require a user reply to the system operator message queue, QSYSOPR, by default.

BS Statement

The BS statement specifies the block size that PowerExchange uses when dynamically allocating target data sets on z/OS if you do not provide a block size.

Operating Systems: z/OS

Data Sources: All

Related Statements: DISP, LRECL, RECFM, RELEASE, SPACE, UNIT, and VOLSER

Required: No

Syntax:

```
BS={block_size|0}
```

Value: For the *block_size* variable, enter a number from 0 through 32760. Default is 0, which indicates that z/OS determines the block size.

CAPI_CONN_NAME Statement

The CAPI_CONN_NAME statement specifies the name of the CAPI_CONNECTION statement that PowerExchange uses by default for all data source types when you do not supply a CAPI connection override.

Operating Systems: All

Related Statements: CAPI_CONNECTION and CAPI_SRC_DFLT

Required: No

Syntax:

```
CAPI_CONN_NAME=capi_connection_name
```

Value: For the *capi_connection_name* variable, enter the name of the CAPI_CONNECTION statement that PowerExchange uses when you do not specify a CAPI_CONNECTION override value.

Usage Notes:

- Optionally, you can use the CAPI_SRC_DFLT statement to provide a source-specific default connection. A source-specific default is useful if the DBMOVER configuration file contains many CAPI_CONNECTION statements.
- Informatica recommends that you always specify an override CAPI_CONNECTION statement for a PowerExchange operation, instead of relying on default connection definitions. You can define the following overrides, based on the type of PowerExchange operation:
 - For CDC sessions, specify the **CAPI Connection Name Override** attribute on the PWX CDC application connection.
 - For PowerExchange Condense, specify the CONN_OVR statement in CAPTPARM configuration file.
 - For the PowerExchange Logger for Linux, UNIX, and Windows, specify the CONN_OVR statement in pwxcl.cfg configuration file.
 - For the generation of restart tokens with the DTLUAPPL utility, specify the CONN_OVR statement in the DTLUAPPL control statement.
 - For CAPXRT database row tests in the PowerExchange Navigator, specify the **CAPI Connection Name** value in the **Advance** dialog box.
 - For the generation of restart tokens during a database row test in the PowerExchange Navigator, specify the CONNAME parameter.

- For PowerExchange ODBC connections, specify the DTLCONN_OVR parameter in the odbc.ini file or the DTLCONN_OVR SQL escape sequence override.

RELATED TOPICS:

- [“Consumer API \(CAPI\) Connection Statements” on page 204](#)

CAPI_CONNECTION - AS4J Statement

The AS4J CAPI_CONNECTION statement specifies a named set of parameters that the Consumer API (CAPI) uses to connect to the change stream and control CDC processing for Db2 for i (i5/OS) CDC sources.

Operating Systems: IBM i (formerly i5/OS)

Data Sources: Db2 for i

Related Statements: CAPI_CONNECTION - UOWC, AS400EVENTMSGQ, AS400USRJRNCODE

Required: Yes for Db2 for i CDC

Syntax:

```
CAPI_CONNECTION=( [DLLTRACE=trace_id]
                  ,NAME=capi_connection_name
                  [,TRACE=trace_name]
                  ,TYPE=(AS4J
                        [,ALWCLRPFM={N|Y}]
                        [,ALWLIBRFSH={N|Y}]
                        [,ALWNONRENT={N|Y}]
                        [,ALWPARTIAL={N|Y}]
                        [,AS4JRNEXT={N|Y}]
                        [,EOF={N|Y}]
                        ,INST=instance_name
                        ,JOURNAL=library/journal_name
                        [,LIBASUSER={N|Y|M|P}]
                        [,NOCCUOWSZ=uow_size]
                        [,POLWAIT={seconds|10}]
                        [,STOPIT={CONT|TERM}
                        ,number_exceptions|5])
                  [,UOWRSTANY={Y|N}]
                  )
```

Parameters:

DLLTRACE=trace_id

Optional. User-defined name of the TRACE statement that activates internal DLL tracing for this CAPI. Specify this parameter only at the direction of Informatica Global Customer Support.

NAME=capi_connection_name

Required. Unique user-defined name for this CAPI_CONNECTION statement.

Maximum length is eight alphanumeric characters.

TRACE=trace_name

Optional. User-defined name of the TRACE statement that activates the common CAPI tracing. Specify this parameter only at the direction of Informatica Global Customer Support.

TYPE=(AS4J, ...)

Required. Type of CAPI_CONNECTION statement. For Db2 for i5/OS sources, this value must be AS4J.

ALWCLRPFM={N|Y}

Optional. Controls whether Db2 for i CDC processing stops or continues when PowerExchange encounters changes that result from an i5/OS Clear Physical File Member (CLRPFM) command or SQL TRUNCATE statement that is issued against a Db2 table registered for change data capture. PowerExchange cannot capture the deletions that result from a CLRPFM command or TRUNCATE statement.

Note: IBM i version 7.2 introduced support for SQL TRUNCATE operations on i5/OS tables. If you have an earlier IBM i version, this parameter pertains to CLRPFM commands only.

Enter one of the following options:

- **N.** PowerExchange CDC processing stops when PowerExchange detects changes that are caused by a CLRPFM command or TRUNCATE statement.
- **Y.** PowerExchange ignores the CLRPFM command or TRUNCATE statement and continues CDC processing. The data integrity of the CDC target might be damaged. If you also specify the AS400EVENTMSGQ statement in the DBMOVER configuration file, PowerExchange issues the DTL3002 message to the specified message queue when PowerExchange encounters a journal entry for a CLRPFM command or TRUNCATE statement.

Attention: If you set this parameter to Y, the data integrity of the CDC targets might be damaged. Specify this parameter only at the direction of Informatica Global Customer Support.

If you also specify a message queue in the AS400EVENTMSGQ parameter of the DBMOVER file, PowerExchange writes message DTL3002 to the queue if PowerExchange encounters a journal entry for a CLRPFM command or TRUNCATE statement.

Default is N.

ALWLIBRFSH={N|Y}

Optional. Controls whether you can refresh the CDC interest list after you add or delete libraries with instances of the same Db2 source table, without stopping and restarting real-time CDC sessions. This parameter is useful in the following situation: 1) you have instances of the same Db2 source table (or file) in multiple libraries, 2) the table instances are journaled to the same journal, and 3) you routinely add or delete libraries that contain an instance of the table. Options are:

- **N.** Disables refresh processing of the CDC interest list for library additions or deletions. If you add libraries, PowerExchange does not start change data capture for those libraries until you stop and restart the CDC session. In this case, some changes might be lost. If you delete libraries, the CDC session ends abnormally.
- **Y.** Enables you to use the SNDPWXJRNE command to refresh the CDC interest list for library additions and deletions. The command sends journal entries for add library and delete library requests to the journal, using the PowerExchange default journal entry type codes or the unique codes that you define in the AS400USRJRNCODE statement. In PowerCenter, you must specify the asterisk (*) wildcard for the library name in the **Library/File Override** session property for the real-time session.

If you also specify a message queue in the AS400EVENTMSGQ parameter of the DBMOVER file, PowerExchange writes message DTL3003I for an *ADDLIB (add library) event and message DTL3004I for a *DLTLIB (delete library) event to the queue.

Default is N.

ALWNONRENT={N|Y}

Optional. Controls whether PowerExchange generates a next receiver (NR) journal entry when i5/OS does not provide one in the journal receiver. PowerExchange requires an NR entry to switch to the

next journal receiver in the chain when the current one becomes full. However, i5/OS might not provide an NR entry under certain conditions, such as when a high level of transaction activity is occurring at the time a journal receiver switch is required. If i5/OS does not provide an NR entry, PowerExchange CDC processing ends unless you enable PowerExchange to generate an NR entry. Options are:

- **N.** Do not generate an NR entry. If the i system does not provide an NR entry, CDC processing ends abnormally.
- **Y.** Generate an NR entry so that CDC processing can continue when the i system does not provide an NR entry. The generated NR entry is for PowerExchange internal use only and is not written to the actual journal receiver.

Default is N.

ALWPARTIAL={N|Y}

Optional. Controls whether PowerExchange processes journal receivers in partial status.

Enter one of the following options:

- **N.** PowerExchange fails processing if a journal receiver is in partial status.
- **Y.** PowerExchange processes journal receivers in partial status.

Default is N.

Attention: If you specify Y for this parameter, you might compromise the data integrity of the change data being extracted because required changes might be unavailable. Specify this parameter only at the direction of Informatica Global Customer Support.

AS4JRNEXIT={N|Y}

Optional. Controls whether PowerExchange uses an exit program installed at the Delete Journal Receiver exit point, QIBM_QJO_DLT_JRNRCV, to prevent the deletion of journal receivers being processed for CDC.

Enter one of the following options:

- **N.** PowerExchange does not lock journal receivers that it is processing.
- **Y.** PowerExchange locks any journal receiver that it is processing so that the journal receiver cannot be deleted from the i system. PowerExchange records the journal receivers that it is processing in a lock file called PWXJRNLCCKP in the CONDLIB library. When PowerExchange switches to the next journal receiver on the chain, PowerExchange removes the record for the previous journal receiver from the lock file.

If you also specify a message queue in the AS400EVENTMSGQ parameter of the DBMOVER file, PowerExchange writes the DTL3001 message to the specified queue for each journal receiver that a change data extraction successfully processes.

Default is N.

EOF={N|Y}

Optional. Controls whether PowerExchange stops change data extractions when the end-of-log (EOL) is reached.

Because this parameter affects all users of the AS4J CAPI_CONNECTION statement, Informatica recommends that you use one of the following alternative methods to stop change data extractions at EOL:

- For CDC sessions that use real-time extraction mode, specify 0 for the **Idle Time** attribute of the PWX DB2i5OS CDC Real Time application connection.
- For PowerExchange Condense, specify 1 for the COLL_END_LOG statement in the CAPTPARM configuration member.
- For CDC sessions that use ODBC connections, specify 0 for the WAITTIME parameter in the ODBC data source.

If you use the EOF parameter, enter one of the following options:

- **N.** PowerExchange does not stop change data extractions when EOL is reached.
- **Y.** PowerExchange stops change data extractions when EOL is reached.

Default is N.

INST=instance_name

Required. User-defined name for the source instance. This name must match the name that you specify in the **Collection Identifier** property of the registration group.

If you are running PowerExchange Condense, this name must also match the name specified in DBID parameter in the CAPTPARM member.

JOURNAL=library/journal_name

Required. Library name and journal name for the journal that contains change data for registered tables and from which PowerExchange extracts change data.

Optionally, you can override this journal specification by using one of the following methods:

- For real-time extraction CDC sessions, use the **Journal Name** on the PWX CDC application connection.
- For PowerExchange Condense, use the JRNL statement in the CAPTPARM configuration member.
- For CAPXRT database row tests, use the **AS400 Journal/Library** property in the **CAPXRT Advanced Parameters** dialog box in the PowerExchange Navigator.
- For real-time extraction CDC sessions that use ODBC connections, use the DTLJRNL parameter for the ODBC data source.

LIBASUSER={N|Y|M|P}

Optional. Controls the type of value that PowerExchange writes to the DTL__CAPXUSER field of each change record.

Options are:

- **N.** Writes the user ID of the user who made the change.
- **Y.** Writes the library name and file name of the file where the change was made.

If you set the ALWLIBRFSH parameter to Y, Informatica recommends that you specify LIBASUSER=Y. This setting enables you to identify the libraries that contain the table instances for which changes are extracted.

- **M.** Writes the library name, file name, and data member name of the file where the change was made.

- **P.** Writes the name of the program that made the change.

Default is N.

NOCCUOWSZ=*uow_size*

Optional. Size, in number of records, of the UOW that PowerExchange creates when reading change records from journal receivers that were created without commitment control.

For the *uow_size* variable, enter a number from 1 through 50000.

If you do not specify commitment control when making changes to Db2 for i tables, PowerExchange creates a UOW for each change record. Use this parameter to increase the size of the UOWs that PowerExchange creates. PowerExchange and PowerCenter process larger UOWs more efficiently. Larger UOWs also reduce commit activity on the targets of the change data.

Because this parameter affects all users of this CAPI_CONNECTION statement, Informatica recommends that you use the **Minimum Rows Per commit** attribute on the PWX DB2i5OS CDC Real Time application connection to specify UOW size values for CDC sessions.

PowerExchange ignores this parameter for change records created with commitment control.

POLWAIT={*seconds*|10}

Optional. The number of seconds that PowerExchange waits after reaching the end of a journal receiver before checking for change data in that journal receiver.

For the *seconds* variable, enter a number from 0 through 10. Default is 10.

STOPIT={({*CONT*|*TERM*}),*number_exceptions*|5}

Optional. Controls whether an extraction session stops after it encounters exceptions when processing change data from the journal. An exception can be one of the following:

- An after image change without a before image change
- An attempt to delete a record that has no data

For the first positional parameter, enter one of the following options:

- **CONT.** The extraction session continues to run after the specified number of exceptions.
- **TERM.** The extraction session ends after the specified number of exceptions.

Default is CONT.

For the second positional parameter, *number_exceptions*, enter the number of exceptions after which PowerExchange takes the action defined in the first positional parameter of the statement. The value 0 indicates that the extraction continues to run but does not report any exceptions. Valid values are 0 through 255. Default is 5.

UOWRSTANY={*Y*|N}

This parameter is deprecated. Y is the default value. When PowerExchange extraction sessions cold start, they skip any partial UOW for which the start-UOW point is on a previous journal receiver. The sessions begin reading change data with the next UOW in the current journal receiver.

If you omit this parameter, the default behavior is the same as if Y were specified. If you enter N, PowerExchange ignores the N value and uses Y.

CAPI_CONNECTION - CAPX Statement

The CAPX CAPI_CONNECTION statement specifies a named set of parameters that the Consumer API (CAPI) uses for continuous extraction of change data from PowerExchange Logger for Linux, UNIX, and Windows log files.

Operating Systems: Linux, UNIX, and Windows

Required: Yes for continuous extraction mode

Syntax:

```
CAPI_CONNECTION=([DLLTRACE=trace_id]
                 ,NAME=capi_connection_name
                 [,TRACE=trace_name]
                 ,TYPE=(CAPX
                     ,DFLTINST=instance_name
                     [,FILEWAIT={seconds|1}]
                     [,NOSEQVAL={N|Y}]
                     [,RSTRADV=seconds]
                     [,RSTRANMODE={N|Y}]
                     [,VALIDATEREGS={N|Y}]
                 )
            )
```

Parameters:

DLLTRACE=trace_id

Optional. User-defined name of the TRACE statement that activates internal DLL tracing for this CAPI. Specify this parameter only at the direction of Informatica Global Customer Support.

NAME=capi_connection_name

Required. Unique user-defined name for this CAPI_CONNECTION statement.

Maximum length is eight alphanumeric characters.

TRACE=trace_name

Optional. User-defined name of the TRACE statement that activates the common CAPI tracing. Specify this parameter only at the direction of Informatica Global Customer Support.

TYPE=(CAPX, ...)

Required. Type of CAPI_CONNECTION statement. For continuous extraction mode, this value must be CAPX.

DFLTINST=instance_name

Required. A source instance identifier that is specified for the registration group. This value must match the DBID value that is specified in the PowerExchange Logger configuration file.

To determine this value, view the registration group properties in the PowerExchange Navigator. Depending on the source type, enter one of the following values:

- For Adabas, Db2 for i (i5/OS), Db2 for z/OS, MySQL, Oracle, PostgreSQL, and VSAM, enter the name that is displayed in the **Instance** field for the registration group in the **Resource Inspector**.
- For Datacom, enter the name of the Multi-User Facility (MUF) in the **MUF Name** field.
- For a Db2 source on Linux, UNIX, or Windows, enter the name of the database in the **Database** field.
- For an IDMS log-based source, enter the name of the database in the **DB Name** field.
- For an IMS source, enter the recon identifier for the database in the **RECON ID** field.

- For Microsoft SQL Server, this value depends on whether you specify the optional DISTSRV and DISTDB parameters in the PowerExchange Logger configuration file:
 - If you specify the DISTSRV and DISTDB parameters, enter the DBID name that you use as the collection identifier for all of the registrations. This name overrides the instance name that is associated with the individual registrations.
 - If you do not specify the DISTSRV and DISTDB parameters, enter the value that the PowerExchange Navigator generates and displays in the **Instance** field of the **Resource Inspector** for the registration group. The generated value is composed of the first four characters of the database name followed by a generated number, which starts at 000.

Maximum length is eight alphanumeric characters.

FILEWAIT={seconds|1}

Optional. The number of seconds that PowerExchange waits before checking for new PowerExchange Logger log files.

For the *seconds* variable, enter a number from 1 through 86400. Default is 1.

NOSEQVAL={N|Y}

Optional. If you receive error message PWX-36944 after starting a CDC session, the sequence token that PWXPC passed to PowerExchange is earlier than the sequence token that is recorded in the PowerExchange Logger CDCT file. If you want the session to continue and you can tolerate some data loss, you can set this parameter to Y. The Log Reader then begins extracting the earliest available data in the log files. With the default value of N, the session fails.

RSTRADV=seconds

Optional. The time interval, in seconds, that PowerExchange waits before advancing the restart and sequence tokens for a registered data source during periods when UOWs contain no changes of CDC interest for a data source. When the wait interval expires, PowerExchange returns the next committed "empty UOW," which includes only updated restart information.

For the *seconds* variable, enter a number from 0 through 86400. No default value is provided. A value of 0 causes PowerExchange to return an empty UOW after each UOW processed. Consequently, a value of 0 can degrade performance.

PowerExchange resets the wait interval to 0 when one of the following events occur:

- PowerExchange completes processing a UOW that includes changes of CDC interest.
- PowerExchange returns an "empty UOW" because the RSTRADV wait interval expired without any new changes of CDC interest having been received.

For example, if you specify 5, PowerExchange waits 5 seconds after it completes processing the last UOW or after the previous RSTRADV interval expires. Then PowerExchange returns the next committed "empty UOW" that includes the updated restart information and resets the wait interval to 0.

If you specify the RSTRADV parameter with any valid value, PowerExchange always advances the restart and sequence tokens when the Log Reader reaches the end of a Logger log file, even if the RSTRADV interval has not expired. This behavior ensures that restart and sequence tokens are advanced even when the CDC session run time is shorter than the RSTRADV interval. This situation is most likely to occur if you have source tables that have a low level of update activity.

If you do not specify RSTRADV, PowerExchange does not advance restart and sequence tokens for a registered source during periods when PowerExchange receives no changes of interest. In this

case, when PowerExchange warm starts, it reads all changes, even those not of CDC interest, from the restart point.

For Db2 for i (i5/OS) sources, Informatica recommends that you use this parameter if the change records that PowerExchange reads from i5/OS journal receivers are created under commitment control. If the change records are created without commitment control, do not specify this parameter.

RSTRANMODE={N|Y}

Optional. If you are migrating from real-time extraction mode to continuous extraction mode for CDC sessions that have not previously used the PowerExchange Logger for Linux, UNIX, and Windows, set this parameter to Y. This setting enables PowerExchange to convert restart token information to the format that PowerExchange Logger requires, when CDC sessions run. Retain the Y setting until PowerExchange completes converting the restart tokens for all registered source tables and all CDC sessions that use this CAPX CAPI_CONNECTION. Then, set this parameter to N or delete it. Default is N.

VALIDATEREGS={N|Y}

Optional. Controls whether the PowerExchange extraction of data from PowerExchange Logger for Linux, UNIX, and Windows ends or continues if it detects a source entry in the CDC interest list for which the PowerExchange Logger is *not* capturing change data. In this situation, the extraction process issues message PWX-36947, either as error message or informational message.

Options are:

- **Y.** The PowerExchange Logger extraction process issues message PWX-36947 as an error message and then ends.
- **N.** The PowerExchange Logger extraction process issues message PWX-36947 as an informational message and continues processing.

Default is N.

CAPI_CONNECTION - LRAP Statement

The LRAP CAPI_CONNECTION statement specifies a named set of parameters that the Consumer API (CAPI) uses for the Log Read API (LRAPI) component of the PowerExchange Logger for z/OS.

The LRAPI connects to the PowerExchange Logger to read change data for the address space that is extracting that data, such as the PowerExchange Listener address space.

Data Sources: Adabas, CA Datacom/DB, CA IDMS/DB, Db2 for z/OS, IMS, and VSAM

Related Statements: CAPI_CONNECTION - UOWC

Required: Yes for z/OS CDC

Syntax:

```
CAPI_CONNECTION=( [DLLTRACE=trace_id]
                  ,NAME=capi_connection_name
                  [,TRACE=trace_name]
                  ,TYPE=(LRAP
                        ,AGENT=agent_id
                        ,LOG=logger_id
                        [,EOF={Y|N}]
                        [,FUZZYRSTART={Y|N}]
                        [,UIDFMT={UID|CONN|CORR|CTYPE|PLAN|ALL}]
                        [,UIDFMTIMS={UID|PSB|ALL}]
                  )
                )
```

Parameters:

DLLTRACE=trace_id

Optional. A user-defined name for the TRACE statement that activates internal DLL tracing for this CAPI. Specify this parameter only at the direction of Informatica Global Customer Support.

NAME=capi_connection_name

Required. A unique user-defined name for this CAPI_CONNECTION statement.

Maximum length is eight alphanumeric characters.

TRACE=trace_name

Optional. A user-defined name for the TRACE statement that activates the common CAPI tracing. Specify this parameter only at the direction of Informatica Global Customer Support.

TYPE=(LRAP, ...)

Required. Type of CAPI_CONNECTION statement. For the LRAPI, this value must be LRAP.

AGENT=agent_id

Required. The PowerExchange Agent ID. This value must match the value in the AGENTID parameter of the EDMSDIR module. PowerExchange reads the EDMSDIR module from the EDMPARMS DD statement, or if this statement is not specified, from the STEPLIB or JOBLIB DD statement.

Maximum length is four alphanumeric characters.

LOG=logger_id

Required. The PowerExchange Logger ID. This value must match the value specified in the LOGGER parameter of the EDMSDIR module.

Maximum length is four alphanumeric characters.

EOF={N|Y}

Optional. Controls whether PowerExchange stops change data extractions after reaching the end-of-log (EOL).

Options are:

- **N.** PowerExchange does not stop change data extractions when EOL is reached.
- **Y.** PowerExchange stops change data extractions when EOL is reached.

Default is N.

Because this parameter affects all users of the LRAP CAPI_CONNECTION statement, Informatica recommends that you use one of the following alternative methods to stop change data extractions at EOL:

- For CDC sessions that use real-time extraction mode, enter 0 for the **Idle Time** attribute on the PWX DB2zOS CDC Real Time application connections.
- For PowerExchange Condense, enter 1 in the COLL_END_LOG statement in the CAPTPARM configuration member.
- For CDC sessions that use ODBC connections, enter 0 for the WAITTIME parameter in the ODBC data source.

FUZZYRSTART={Y|N}

Optional. Controls whether the restart2 token in the restart information that the PowerExchange Client for PowerCenter sends to the PowerExchange Logger for z/OS for a CDC session points to a begin-uow record or any RBA position that you specify in the restart2 token.

This parameter also pertains to the Restart Token 2 value that you specify for PowerExchange Navigator CAPX and CAPXRT database row tests.

Options are:

- **Y.** Allow the restart2 token to point to any position in the PowerExchange Logger logged data. This option enables you to skip a problematic record.
- **N.** Require the restart2 token to point to a begin-uow record in the PowerExchange Logger logged data.

Default is Y.

UIDFMT={UID|CONN|CORR|CTYPE|PLAN|ALL}

Optional. For Db2 for z/OS CDC sources, controls the type of value that PowerExchange returns in the generated DTL__CAPXUSER field in each change record. Options are:

- **UID.** Returns the user ID of the user who made the change.
- **CONN.** Returns the Db2 connection identifier at the time the change was made.
- **CORR.** Returns the Db2 correlation identifier at the time the change was made.
- **CTYPE.** Returns the Db2 connection type at the time the change was made.
- **PLAN.** Returns the Db2 plan name at the time the change was made.
- **ALL.** Returns all of the information supplied by the other options. PowerExchange provides this information in a colon-delimited list in the following format:

userid:plan_name:correlation_id:connection_id:connection_type

Default is UID.

Restriction: You can specify only one option.

UIDFMTIMS={UID|PSB|ALL}

Optional. For IMS synchronous CDC data sources, controls the type of value that PowerExchange uses to populate the generated DTL__CAPXUSER column in each change record. Options are:

- **UID.** Returns the user ID of the user who made the IMS change.
- **PSB.** Returns the IMS program specification block (PSB) name.
- **ALL.** Returns both the user ID and PSB name in the format *userid:psbname*.

Default is UID.

Restriction: You can specify only one option.

CAPI_CONNECTION - MSQL Statement

The MSQL CAPI_CONNECTION statement specifies a named set of parameters that the Consumer API (CAPI) uses to connect to the change stream and control CDC processing for Microsoft SQL Server sources.

Operating Systems: Windows

Data Sources: Microsoft SQL Server

Required: Yes for Microsoft SQL Server CDC

Syntax:

```
CAPI_CONNECTION=( [DLLTRACE=trace_id]
                  ,NAME=capi_connection_name
                  [,TRACE=trace_name]
```

```

, TYPE=(MSQL
      ,DISTDB=distribution_database
      ,DISTSRV=distribution_server
      [, BATCHSIZE=number]
      [, DWFLAGS={ flag1flag2flag3flag4 | NNNN } ]
      [, ENABLELWM={ N | Y } ]
      [, EOF={ N | Y } ]
      [, MEMCACHE={ cache_size | 256 } ]
      [, MULTIPUB={ N | Y } ]
      [, POLWAIT={ seconds | 1 } ]
      [, RECONNTRIES={ number | 12 } ]
      [, RECONNWAIT={ seconds | 5 } ]
      [, RSTRADV=seconds]
      [, SQLNOLOCK={ N | Y } ]
      [, UIDFMT={ DBNAME | NONE } ]
)

```

Parameters:

DLLTRACE=*trace_id*

Optional. User-defined name of the TRACE statement that activates internal DLL tracing for this CAPI. Specify this parameter only at the direction of Informatica Global Customer Support.

NAME=*cap_i_connection_name*

Required. Unique user-defined name for this CAPI_CONNECTION statement.

Maximum length is eight alphanumeric characters.

TRACE=*trace_name*

Optional. User-defined name of the TRACE statement that activates the common CAPI tracing. Specify this parameter only at the direction of Informatica Global Customer Support.

TYPE=(MSQL, ...)

Required. Type of CAPI_CONNECTION statement. For Microsoft SQL Server sources, this value must be MSQL.

DISTDB=*distribution_database_name*

Required. The name of the distribution database.

DISTSRV=*distribution_database_server*

Required. The network name of the server that hosts the distribution database. This name is different from the network name of the SQL Server publication instance if the distribution database resides on a different instance.

Note: If the database server uses a port number other than the default port number of 1433, append the non-default port number to the server name by using the following format:

`\server_name,port_number\`. Otherwise, capture processing fails.

BATCHSIZE=*number*

Optional. The number of rows from which PowerExchange captures change data before closing the cursor and then reopening it. This parameter allows resources to be released periodically to reduce the capture processing load on system memory and to reduce temporary tables in the tempdb database. Valid values are 0 through 2147483647. No default is provided.

Specify this parameter only at the direction of Informatica Global Customer Support. It can degrade CDC performance because PowerExchange issues the data read query more often.

DWFLAGS={*flag1flag2flag3flag4*|NNNN}

Optional. Series of four positional parameters that control whether processing stops or continues when data loss, truncation, schema changes, or unrecognized transaction log records occur.

Specify this statement only at the direction of Informatica Global Customer Support.

Enter the following positional parameters:

- *flag1*. Controls whether PowerExchange stops a change data extraction when PowerExchange retrieves data of an unexpected length from the distribution database. Enter Y to continue processing or enter N to stop processing.
- *flag2*. Controls whether PowerExchange stops a change data extraction when it detects a schema change. Enter Y to continue processing or enter N to stop processing.
- *flag3*. Controls whether PowerExchange stops a change data extraction when PowerExchange does not find the requested start sequence in the transaction log. Enter Y to continue processing or enter N to stop processing.
- *flag4*. Controls whether PowerExchange stops a change data extraction when PowerExchange finds an unrecognized record in the transaction log. Enter Y to continue processing after error message PWX-15742 or enter N to stop processing.

Default is NNNN, which indicates none of the parameters are set.

ENABLELWM={*N*|*Y*}

Optional. When you use the PowerExchange Logger for Linux, UNIX, and Windows, controls whether the PowerExchange consumer API (CAPI) connection process deletes data read from the SQL Server distribution database after the data has been hardened to PowerExchange Logger log files or after the PowerExchange publication expiry time has elapsed. You can use this parameter to improve distribution database performance and to prevent the distribution database from growing too large in size when the PowerExchange Logger is in use.

Enter one of the following options:

- **N**. The distribution database cleanup job deletes data from the distribution database after the expiry time for the PowerExchange publications elapses. This option might degrade the performance of the distribution-database cleanup job and cause excessive growth of the distribution database.

- **Y.** The CAPI connection process deletes processed data from the distribution database after the data has been hardened to the PowerExchange Logger log files. After a log file switch, the PowerExchange Logger sends a low water marker to the CAPI connection process to identify the last end UOW prior to the file switch. At the end of the next capture cycle, after the CAPI connection process has read to the end of the available data in the distribution database, the CAPI deletes all of the processed data for the PowerExchange publications up to and including the low water mark data from the distribution.dbo.MSrepl_commands table in the distribution database.

Note: The user ID under which the PowerExchange Logger runs must have delete authority on the MSrepl_commands table.

This option can help improve distribution-database performance and control distribution-database size. However, if the SQL Server Log Reader Agent is writing very large UOWs to the distribution database when the CAPI connection processes the low water mark data, the performance of the distribution database might be temporarily degraded because the CAPI connection process must wait for a lock on the MSrepl_commands table.

Note: If you run multiple extractions against a single distribution database for different publication databases and use ENABLELWM=Y for one CAPI connection and ENABLELWM=N with a RSTRADV value for another CAPI connection, PowerExchange might issue error message PWX-15756 for the connection with ENABLELWM=N. The message incorrectly reports that change data has been lost. To suppress this error, add the DWFLAGS=NNYN parameter to MSQL CAPI_CONNECTION statement.

Default is N.

EOF={N|Y}

Optional. Controls whether PowerExchange stops change data extractions when the end-of-log (EOL) is reached.

Enter one of the following options:

- **N.** PowerExchange does not stop change data extractions when the EOL is reached.
- **Y.** PowerExchange stops change data extractions when the EOL is reached.

Default is N.

Because this parameter affects all users of the MSQL CAPI_CONNECTION statement, Informatica recommends that you use one of the following alternative methods to stop change data extractions at the EOL:

- For CDC sessions that use real-time extraction mode, enter 0 for the **Idle Time** attribute of the PWX MSSQL CDC Real Time application connection.
- For the PowerExchange Logger for Linux, UNIX, and Windows, enter 1 for the COLL_END_LOG statement in the pwxcl.cfg configuration file.
- For CDC sessions that use ODBC connections, enter 0 for the WAITTIME parameter in the ODBC data source.

MEMCACHE={cache_size|256}

The maximum size, in kilobytes, of the memory cache that stores change data for a single SQL operation that is captured from the SQL Server distribution database. The memory cache stores the full row image, which can include both the before image and after image and any LOB data.

Valid values are 0 through 2147483647. Default is 256. If you enter 0, the default value is used.

MULTIPUB={N|Y}

Optional. Indicates whether you capture change data from the distribution database for articles in a single publication database or in multiple publication databases. This option can affect the performance of CDC processing in real time extraction mode and in continuous extraction mode with the PowerExchange Logger for Linux, UNIX, and Windows.

Enter one of the following options:

- **N.** Specify this option if you capture change data for articles in a single publication database. Informatica recommends this option in this scenario because it causes PowerExchange to extract changes much more efficiently. It can also help reduce resource usage.
- **Y.** Use this option to extract change data for articles in multiple publication databases in a single CDC session or in a single PowerExchange Logger for Linux, UNIX, and Windows pass. If you do not use this option in this scenario, extraction processing fails with message PWX-15757.

This option might cause change records to be written to the distribution database more slowly. To improve performance, add the following index to the distribution database:

```
USE [distribution]
GO
/***** Object: Index [IX_MSrepl_transactions] Script Date: 03/31/2012
11:56:07 *****/
CREATE NONCLUSTERED INDEX [IX_MSrepl_transactions] ON [dbo].[MSrepl_transactions]
(
    [entry_time] ASC,
    [publisher_database_id] ASC,
    [xact_seqno] ASC,
    [xact_id] ASC
) WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = ON, SORT_IN_TEMPDB = OFF,
IGNORE_DUP_KEY = OFF, DROP_EXISTING = OFF, ONLINE = OFF, ALLOW_ROW_LOCKS =
ON, ALLOW_PAGE_LOCKS = ON) ON [PRIMARY]
GO
```

Default is Y.

You can change the MULTIPUB setting after capturing changes. To maintain data integrity, follow the procedure for changing the MULTIPUB parameter setting in the *PowerExchange CDC Guide for Linux, UNIX, and Windows*. If you use the PowerExchange Logger for Linux, UNIX, and Windows and change the setting from Y to N, you must cold start the PowerExchange Logger.

POLWAIT={seconds|1}

Optional. The maximum number of seconds that PowerExchange waits after reaching the end of log before polling the source database for more change data.

For Microsoft SQL Server sources, the polling frequency also depends on the PowerExchange Logger NO_DATA_WAIT2 parameter, or if you do not use the PowerExchange Logger, the polling frequency depends on the PWX Latency attribute on the PWX CDC application connection. If the NO_DATA_WAIT2 or PWX Latency value is less than the POLWAIT value, the lesser value takes precedence. In this case, PowerExchange polls the source more frequently than expected based on the POLWAIT parameter only.

Valid values are 1 through 2147483647. Default is 1.

RECONNTRIES={number|12}

The maximum number of times that PowerExchange tries to reconnect to the Microsoft SQL Server database after the connection is dropped. Use this parameter in conjunction with the RECONNWAIT

parameter if you get the following ODBC connection error and want to improve connection resiliency:

```
PWX-15790 ODBC driver for Microsoft SQL Server returned error [08S01]
[Informatica][ODBC SQL Server Wire Protocol driver]Unexpected Network Error.
ErrNum = 10054.
```

Valid values are 0 or any positive number. A value of 0 results in no connection retries. Default is 12.

RECONNWAIT={seconds|5}

The number of seconds that PowerExchange waits before any attempt to reconnect to a Microsoft SQL Server database after the connection has been dropped. Use this parameter in conjunction with the RECONNTRIES parameter if you get the PWX-15790 message for an ODBC driver error and want to improve connection resiliency.

Valid values are 0 through 3600. A value of 0 results in no waiting between connection retries. Default is 5.

RSTRADV=seconds

Time interval, in seconds, that PowerExchange waits before advancing restart and sequence tokens for a registered data source during periods when UOWs do not include any changes of interest for the data source. When the wait interval expires, PowerExchange returns the next committed "empty UOW," which includes only updated restart information.

PowerExchange resets the wait interval to 0 when one of the following events occur:

- PowerExchange completes processing a UOW that includes changes of interest.
- PowerExchange returns an empty UOW because the wait interval expired without PowerExchange receiving any changes of interest.

Valid values 0 through 86400. No default is provided.

For example, if you specify 5, PowerExchange waits 5 seconds after it completes processing the last UOW or after the previous wait interval expires. Then PowerExchange returns the next committed empty UOW that includes the updated restart information and resets the wait interval to 0.

If you do not specify RSTRADV, PowerExchange does not advance restart and sequence tokens for a registered source during periods when PowerExchange receives no changes of interest. In this case, when PowerExchange warm starts, it reads all changes, including those not of interest for CDC, from the restart point.

Warning: A value of 0 can degrade performance. In addition to the UOWs that contain changes for registered sources of interest, PowerExchange returns an empty UOW for every UOW that does not contain changes for the registered sources of interest.

SQLNOLOCK={N|Y}

Optional. Controls whether PowerExchange SELECT statements use the NOLOCK hint when querying the SQL Server distribution database for change data capture. The NOLOCK hint can avoid lock contention with the SQL Server utilities but might cause PowerExchange to miss some change records.

Enter one of the following options:

- **N.** PowerExchange SELECT queries that retrieve data from the distribution database do *not* use the NOLOCK hint. If locks are held on some change records, PowerExchange queries cannot retrieve the data until the locks are released. With this setting, PowerExchange queries might take longer to complete. However, no changes are skipped and data integrity is preserved. Use this option only when the MULTIPUB parameter is set to Y.
- **Y.** PowerExchange SELECT queries that retrieve data from the distribution database use the NOLOCK hint. Use this option only when the MULTIPUB parameter is set to N. If the MULTIPUB parameter is set to Y, SQL Server might use allocation order scans to retrieve data for PowerExchange queries, which can result in missed change data and data corruption.

Tip: Instead of using SQLNOLOCK=Y, Informatica recommends that you set the isolation level for the distribution database to READ_COMMITTED_SNAPSHOT ON to avoid data integrity problems.

Default is **N** if MULTIPUB is set to Y, or **Y** if MULTIPUB is set to N.

UIDFMT={DBNAME |NONE}

Optional. Controls the type of value that PowerExchange uses to populate the generated DTL__CAPXUSER column in each change record. Options are:

- **DBNAME.** Returns the Microsoft SQL Server publication database name.
- **NONE.** Returns a null because a user ID is not available.

Default is NONE.

CAPI_CONNECTION - MYSQL Statement

The MYSQL CAPI_CONNECTION statement specifies a named set of parameters that the Consumer API (CAPI) uses to connect to the change stream and control CDC processing for MySQL sources.

Operating Systems: Linux and Windows

Data Sources: MySQL

Required: Yes for MySQL CDC

Syntax:

```
CAPI_CONNECTION=(NAME=capi_connection_name
[,DLLTRACE=trace_id]
,TYPE=(MYSQL
,SERVER={database_server|localhost}
,CATSCHEMA=catalog_schema_name
[,CATSERVER=catalog_server]
[,CATUSERNAME=catalog_user_name]
[,CATPASSWORD=encrypted_catalog_user_password]
[,CATPASSWORD=catalog_user_password]
[,MYSQLBINLOG=path/binlog_file [option1 option2 ...]]
[,ONDATA TRUNC={WARN|FAIL}}
[,ONTABLEDDL={WARN|WARN-UNTIL-EOL|FAIL}}
[,ONTABLETRUNC={WARN|FAIL}}
[,RECONNTRIES={reconnection_attempts|12}}
[,RECONNWAIT={seconds|5}}
[,ROWMEMMAX=bytes]
[,RSTRADV=seconds]
[,UOWREADAHEAD={minimum_transactions|5},{maximum_transactions|10}}
)
```

Parameters:

NAME=capi_connection_name

Required. A unique user-defined name for this CAPI_CONNECTION statement.

Maximum length is eight alphanumeric characters.

DLLTRACE=trace_ID

Optional. The user-defined name of the TRACE statement that activates internal DLL tracing for this CAPI. Specify this parameter only at the direction of Informatica Global Customer Support.

TYPE=(MYSQL, ...)

Required. The type of CAPI_CONNECTION statement. For MySQL sources, this value must be MYSQL.

SERVER={server_name|localhost}[port_number]

Required. The host name or IP address of the MySQL server where the MySQL source database runs. You can enter "localhost" if the MySQL server and PowerExchange Listener run locally on the same machine.

If you want the source server to listen on a port number other than the default port, append the port number to the server name.

CATSERVER

Optional. The name of the server that contains the DDL-updated catalog tables that store MySQL source table definitions.

Default is the MySQL source server name. You can use the default if you created the catalog tables and installed the MySQL ODBC drivers on the source server.

CATSCHEMA

Required. The name of the schema for the DDL catalog tables.

CATUSERNAME

Optional. A user name that is used to connect to the catalog server.

Default is the MySQL source user name if the catalog tables exist on the source server.

CATEPASSWORD

Optional. The encrypted password of the specified catalog user. Specify either CATEPASSWORD or CATPASSWORD but do not specify both parameters.

Default is the password of the MySQL source user if the catalog tables exist on the source server.

CATPASSWORD

Optional. The clear-text password of the specified catalog user. Specify either CATPASSWORD or CATEPASSWORD but do not specify both parameters.

Default is the password of the MySQL source user if the catalog tables exist on the source server.

MYSQLBINLOG=path/binlog_file_name [option1 option2...]

Optional. If the MySQL server is not installed on the local host, specify the path to the mysqlbinlog utility in this parameter or in the PATH environment variable. PowerExchange uses the mysqlbinlog utility to read change events from the MySQL binary log. You can optionally include any of the options that the mysqlbinlog utility supports, such as the --ssl- options. PowerExchange passes the options to the utility when the CDC session runs. For information about the utility options, see the *MySQL Reference Manual*.

If any part of the MYSQLBINLOG value includes spaces, you must enclose the entire MYSQLBINLOG value in double-quotation marks ("). Furthermore, if any part of the *path\binlog_file_name* value or an option includes a space, escape the value with the backslash (\) character and also enclose the value in double-quotation marks, for example:

```
MYSQLBINLOG="\c:\bin\test dir\mysqlbinlog.exe\" --ssl-mode required "
```

If the *path\binlog_file_name* value begins with the hyphen (-) character, the entire value is treated as a utility option, as if no *path/file_name* value was specified.

ONDATATRUNC={WARN|FAIL}

Optional. Indicates whether PowerExchange issues a warning message and continues processing or ends abnormally when it needs to truncate data from MySQL columns that have a mediumblob, longblob, mediumtext, longtext, enum, json, set, varbinary, or varchar datatype and are longer than 98,304 bytes.

Default is FAIL.

ONTABLEDDL={WARN|WARN-UNTIL-EOL|FAIL}

Optional. Indicates whether PowerExchange issues a warning message and continues processing or ends abnormally when it encounters a DDL record for a source that is not consistent with the source registration. Options are:

- WARN. Issue a warning message and continue capture processing.
- WARN-UNTIL-EOL. Issue a warning message and continue capture processing until the end-of-log (EOL). Then issue an error message and terminate capture processing.
- FAIL. Issue an error message and terminate capture processing.

Default is WARN-UNTIL-EOL.

ONTABLETRUNC={WARN|FAIL}

Optional. Indicates whether PowerExchange issues a warning message and continues processing or ends abnormally when it encounters a TRUNCATE TABLE record in the change stream.

Default is FAIL.

RECONNTRIES={reconnection_attempts|12}

Optional. The maximum number of times that PowerExchange tries to reconnect to the MySQL Server database server after detecting that the database server has shut down or the network connection to the server has been dropped. When PowerExchange and the database server run on the same machine, PowerExchange processes change records to the end of log and then checks whether the server is still running. If PowerExchange connects to a remote database server and the server shuts down or the network connection to the server is severed, the binary log reader connection might be dropped. In this case, you can use this parameter in conjunction with the RECONNWAIT parameter to improve connection resiliency.

Valid values are 0 or any positive number. A value of 0 results in no connection retries. Default is 12.

RECONNWAIT={seconds|5}

Optional. The number of seconds that PowerExchange waits before any attempt to reconnect to a MySQL Server database server after detecting that the server has shut down or the network connection to the server has been dropped. Use this parameter in conjunction with the RECONNTRIES parameter to improve connection resiliency.

Valid values are 0 through 3600. A value of 0 results in no waiting between connection retries. Default is 5.

ROWMEMMAX=maximum_bytes

Optional. The maximum amount of memory, in bytes, that PowerExchange can use to store a row change from a MySQL source table. No default value is available. If no maximum value is provided, PowerExchange does not limit the memory for a captured row change.

RSTRADV=seconds

Optional. The time interval, in seconds, that PowerExchange waits before advancing restart and sequence tokens for a registered data source during periods when UOWs do not include any changes of interest for the data source. When the wait interval expires, PowerExchange returns the next committed "empty UOW," which includes only updated restart information.

PowerExchange resets the wait interval to 0 when one of the following events occur:

- PowerExchange completes processing a UOW that includes changes of interest.
- PowerExchange returns an empty UOW because the wait interval expired without PowerExchange receiving any changes of interest.

Valid values 0 through 86400. No default is provided. A value of 0 disables restart advance processing.

For example, if you specify 5, PowerExchange waits 5 seconds after it completes processing the last UOW or after the previous wait interval expires. Then PowerExchange returns the next committed empty UOW that includes the updated restart information and resets the wait interval to 0.

If you do not specify RSTRADV, PowerExchange does not advance restart and sequence tokens for a registered source during periods when PowerExchange receives no changes of interest. In this case, when PowerExchange warm starts, it reads all changes, including those not of interest for CDC, from the restart point.

UOWREADAHEAD=minimum_transactions_in_queue,maximum_transactions_in_queue

Optional. The minimum number of transactions and the maximum number of transactions that control I/O on the UOW read-ahead queue that PowerExchange uses to prefetch change data. The UOW read-ahead queue is a dynamic queue of buffered transactions that are waiting to be read by the PowerExchange capture process. When the queue reaches the maximum number of transactions, the queue reader thread stops and sleeps until enough transactions have been read from the queue to return it to the maximum value. The number of transactions in the queue might become less than the minimum number in the following cases:

- PowerExchange has read to the end of the binary log and no additional changes are available.
- The transactions in the queue are consumed faster than the changes are read from the binary log.

For the *minimum_transactions* field, the default value is 5 and the valid range of values is 0 through 100. For the *maximum_transactions* field, the default value is 10 and the maximum is 100.

Note: The maximum number of transaction that the queue can hold is 100.

CAPI_CONNECTION - ORAD Statement

The ORAD CAPI_CONNECTION statement specifies a named set of parameters that the Consumer API (CAPI) uses to connect to the change stream and control PowerExchange Express CDC for Oracle processing for Oracle data sources.

Operating Systems: Linux, UNIX, and Windows

Data Sources: Oracle

Related Statements: ORACLEID, ORACLE_CAPTURE_TYPE

Required: Yes, for PowerExchange Express CDC for Oracle

Syntax:

```
CAPI_CONNECTION=( [DLLTRACE=trace_id]
                  ,NAME=capi_connection_name
                  ,TYPE=(ORAD
                        [,EPWD=database_encrypted_password]
                        ,ORACOLL=collection_id
                        [,PARMFILE=express_cdc_configuration_file]
                        [,PASSWORD=database_password]
                        [,USERID=database_user_id]
                        )
                  )
```

Parameters:

DLLTRACE=trace_id

Optional. User-defined name of the TRACE statement that activates internal DLL tracing for this CAPI.

Specify this parameter only at the direction of Informatica Global Customer Support.

NAME=capi_connection_name

Required. Unique user-defined name for this CAPI_CONNECTION statement.

Maximum length is eight alphanumeric characters.

TYPE=(ORAD, ...)

Required. Type of CAPI_CONNECTION statement. For PowerExchange Express CDC for Oracle sources, this value must be ORAD.

EPWD=database_encrypted_password

Optional. An encrypted password that PowerExchange uses to connect to the Oracle source database for PowerExchange Express CDC for Oracle. If you specify the USERID parameter in this statement, you must also specify either the EPWD or PASSWORD parameter.

This encrypted password overrides the EPWD parameter value in the DATABASE statement of the PowerExchange Express CDC for Oracle configuration file and the **Password** value in the CDC session connection attributes. To use one of these other encrypted passwords, do not include the EPWD parameter in the ORAD CAPI_CONNECTION statement.

ORACOLL=collection_id

Required. The collection identifier for the Oracle instance. This value must match the collection ID in the first positional parameter of an ORACLEID statement in the same dbmover.cfg file.

Usually, this value also matches the collection ID that you specify in the registration group for the Oracle instance. If you specify a different collection ID in the registration group, the registration collection ID overrides this ORACOLL value.

PARMFILE=path_and_filename

Optional. The path and file name for the PowerExchange Express CDC for Oracle configuration file, relative to the current working directory. You can use this parameter to override the default path and file name or to remind PowerExchange users of the default path and file name.

If this parameter is not specified, PowerExchange uses *pw_x_home_directory\pwxorad.cfg* by default. The default path is the path in the PWX_HOME environment variable, or if this environment variable is not defined, the default path is the path to the PowerExchange bin directory. If the pwxorad.cfg

file does not exist at the default location and a PARMFILE override is not defined, PowerExchange issues error messages PWX-09951 and PWX-00268 and change capture fails.

PASSWORD=database_password

Optional. A clear text password that PowerExchange uses to connect to the Oracle source database for PowerExchange Express CDC for Oracle. If you specify the USERID parameter in this statement, you must specify either the EPWD or PASSWORD parameter.

This clear-text password overrides the PASSWORD parameter in the DATABASE statement of the PowerExchange Express CDC for Oracle configuration file and the **Password** value in the CDC session connection attributes. To use one of these other passwords, do not include the PASSWORD parameter in the ORAD CAPI_CONNECTION statement.

USERID=database_user_id

Optional. A user ID that PowerExchange uses to connect to the Oracle source database for PowerExchange Express CDC for Oracle. If you specify the USERID parameter in this statement, you must also specify either the EPWD or PASSWORD parameter.

This user ID overrides the USERID parameter in the DATABASE statement of the PowerExchange Express CDC for Oracle configuration file and the **User Name** value in the CDC session connection attributes. To use one of these other user IDs, do not include the USERID parameter in the ORAD CAPI_CONNECTION statement.

Usage Notes:

- You can specify multiple ORAD CAPI_CONNECTION statements in the dbmover.cfg file to capture change data from more than one Oracle instance or to use different parameter settings for the same Oracle instance.
- Define the ORAD CAPI_CONNECTION and ORACLEID statements on the PowerExchange system that must connect to the Oracle source database for change data capture. Usually, the PowerExchange Logger for Linux, UNIX, and Windows runs on this system.
- The database user ID and password or encrypted password can be specified in multiple locations. If you do so, PowerExchange uses the following order of precedence:
 1. The USERID value and the EPWD or PASSWORD value that are specified in the ORAD CAPI_CONNECTION statement in the dbmover configuration file
 2. The USERID value and EPWD or PASSWORD value that are specified in the DATABASE statement in the PowerExchange Express CDC for Oracle configuration file, pwxorad.cfg.
 3. The **User Name** and **Password** values that are specified in the Oracle application connection attributes for the PowerCener CDC session

CAPI_CONNECTION - PG Statement

The PG CAPI_CONNECTION statement specifies a named set of parameters that the Consumer API (CAPI) uses to connect to the change stream and control CDC processing for PostgreSQL sources.

Operating Systems: Windows

Data Sources: PostgreSQL

Required: Yes for PostgreSQL CDC

Syntax:

```
CAPI_CONNECTION=(NAME=capi_connection_name
                  [,DLLTRACE=trace_id]
                  ,TYPE=(PG
```

```

, SERVER={database_server|localhost}[,port]
, DATABASE=database_name
[, ENABLELWM={Y|N}]
[, FETCHLIMIT=maximum_rows_fetched|100]
[, ONDATATRUNC={WARN|FAIL}]
[, ONTABLETRUNC={WARN|FAIL}]
[, RECONNTRIES={reconnection_attempts|12}]
[, RECONNWAIT={seconds|5}]
[, REPLSTORETBL=schema.table_name|public.pwx_repl]
[, RSTRADV=seconds]
)
)

```

Parameters:

NAME=capi_connection_name

Required. A unique user-defined name for this CAPI_CONNECTION statement.

Maximum length is eight alphanumeric characters.

DLLTRACE=trace_ID

Optional. The user-defined name of the TRACE statement that activates internal DLL tracing for this CAPI. Specify this parameter only at the direction of Informatica Global Customer Support.

TYPE=(PG, ...)

Required. The type of CAPI_CONNECTION statement. For PostgreSQL sources, this value must be PG.

SERVER={host_name|localhost}[,port_number]

Required. The host name or IP address of the server where the PostgreSQL source database runs. You can enter "localhost" if the database server and PowerExchange Listener run locally on the same machine.

If you want the source server to listen on a port number other than the default port, you can optionally append the port number to the server name.

DATABASE=database_name

Required. The name of the PostgreSQL source database on the specified server.

ENABLELWM={N|Y}

Optional. When you use the PowerExchange Logger for Linux, UNIX, and Windows, controls whether the PowerExchange consumer API (CAPI) capture process deletes data read from the replication store table after the data has been hardened to PowerExchange Logger log files. You can use this parameter to improve capture performance and to prevent the replication store table from growing too large when the PowerExchange Logger is in use.

Options are:

- **N.** Do not delete processed data from the replication store table after the data has been hardened to the PowerExchange Logger log files. With this option, the replication store table might become very large, causing CDC performance to be degraded.
- **Y.** Delete processed data from the replication store table after the data has been hardened to the PowerExchange Logger log files. After a log file switch, the PowerExchange Logger sends a low water marker (LWM) to the CAPI process to identify the last end UOW prior to the file switch. At the end of the next capture cycle, after the CAPI connection process has read to the end of the available data, the CAPI deletes all of the rows in the replication store table up to the LWM.

Default is N.

FETCHLIMIT

Optional. The maximum number of rows that can be fetched from the replication store table in response to a PowerExchange SELECT call. Valid values are 0 through 1000000. A value of 0 means no maximum limit is in effect.

Default is 100.

ONDATATRUNC={WARN|FAIL}

Optional. Indicates whether PowerExchange issues a warning message and continues processing or ends abnormally when it needs to truncate data from PostgreSQL columns that are longer than 98,304 bytes.

Default is FAIL.

ONTABLETRUNC={WARN|FAIL}

Optional. Indicates whether PowerExchange issues a warning message and continues processing or ends abnormally when it encounters a TRUNCATE TABLE record in the change stream.

Default is FAIL.

RECONNTRIES={*reconnection_attempts*|12}

Optional. The maximum number of times that PowerExchange tries to reconnect to the PostgreSQL database server after a connection is dropped while PowerExchange is retrieving data from the replication slot or reading data from the replication store table. If a connection retry is unsuccessful, PowerExchange waits for number of seconds specified in the RECONNWAIT parameter before trying to reconnect to the database server again. Use this parameter in conjunction with the RECONNWAIT parameter to improve connection resiliency.

Valid values are 0 through 2147483647. A value of 0 results in no connection retries. Default is 12.

RECONNWAIT={*seconds*|5}

Optional. The number of seconds that PowerExchange waits between attempts to reconnect to a PostgreSQL database server. Use this parameter in conjunction with the RECONNTRIES parameter to improve connection resiliency.

Valid values are 0 through 3600. A value of 0 results in no waiting between connection attempts. Default is 5.

REPLSTORETBL={*schema.table_name*|public.pwx_repl}

Optional. The name of the PowerExchange replication store table that contains the change data from the PostgreSQL logical replication slot. PowerExchange reads change data from this table.

If this value is not specified, the default name of public.pwx_repl is used.

RSTRADV=*seconds*

Optional. The time interval, in seconds, that PowerExchange waits before advancing restart and sequence tokens for a registered data source during periods when UOWs do not include any changes of interest for the data source. When the wait interval expires, PowerExchange returns the next committed "empty UOW," which includes only updated restart information.

PowerExchange resets the wait interval to 0 when one of the following events occur:

- PowerExchange finishes processing a UOW that includes changes of interest.
- PowerExchange returns an empty UOW because the wait interval expired without PowerExchange receiving any changes of interest.

Valid values 0 through 86400. No default is provided. A value of 0 disables restart advance processing.

For example, if you specify 5, PowerExchange waits 5 seconds after it completes processing the last UOW or after the previous wait interval expires. Then PowerExchange returns the next committed empty UOW that includes the updated restart information and resets the wait interval to 0.

If you do not specify RSTRADV, PowerExchange does not advance restart and sequence tokens for a registered source during periods when PowerExchange receives no changes of interest. In this case, when PowerExchange warm starts, it reads all changes, including those not of interest for CDC, from the restart point.

CAPI_CONNECTION - UDB Statement

The UDB CAPI_CONNECTION statement specifies a named set of parameters that the Consumer API (CAPI) uses to connect to the change stream and to control CDC processing for DB2 for Linux, UNIX, and Windows sources.

Add this statement to the dbmover.cfg file on the system where DB2 capture registrations reside. This location corresponds to the **Location** node that you specify when defining a registration group. Usually, this location is where the source database resides.

Operating Systems: Linux, UNIX, and Windows

Data Sources: DB2 for Linux, UNIX, and Windows

Related Statements: N/A

Required: Yes for CDC

Syntax:

```
CAPI_CONNECTION=( [DLLTRACE=trace_id]
                  ,NAME=capi_connection_name
                  [,TRACE=trace_name]
                  ,TYPE=(UDB
                        [,AGEOUTPERIOD=minutes]
                        [,CCATALOG={capture_catalog|creator.DTLCCATALOG}]
                        [,DBCONN=database_name]
                        [,EPWD=encrypted_password]
                        [,LARGEOPS=number_of_operations]
                        [,MEMCACHE={cache_size|1024}]
                        [,MONITORINT={minutes|5}]
                        [,PASSWORD=password]
                        [,RSTRADV=seconds]
                        [,SPACEPRI={AUTO|MAX|NONE|nn}]
                        [,THREADING={AUTO|MAX|NONE|nn}]
                        [,UDBSchema=schema]
                        [,UPDINT={seconds|600}]
                        [,UPDREC={records|1000}]
                        [,USERID=user_id]
                        )
                  )
```

Parameters:

AGEOUTPERIOD=minutes

The age, in minutes, at which an outstanding UOW that has no change records of CDC interest will be removed from the calculation of the CDC restart point. The age is calculated as the difference between the start time of the outstanding UOW and the current time.

Use this parameter to prevent CDC failures that can occur if you shut down and then restart capture processing while the transaction is outstanding. After the restart, the DB2 transaction log in which the outstanding UOW started might not be available, causing the PowerExchange DB2 read process to fail.

Valid values are 60 to 43200. No default value is provided.

DLLTRACE=trace_id

Optional. A user-defined name for the TRACE statement that activates internal DLL tracing for this CAPI.

Specify this parameter only at the direction of Informatica Global Customer Support.

NAME=capi_connection_name

Required. A unique user-defined name for this CAPI_CONNECTION statement.

Maximum length is eight alphanumeric characters.

TRACE=trace_name

Optional. A user-defined name for the TRACE statement that activates the common CAPI tracing.

Specify this parameter only at the direction of Informatica Global Customer Support.

TYPE=(UDB, ...)

Required. The type of CAPI_CONNECTION statement. For DB2 for Linux, UNIX, and Windows sources, this value must be UDB.

CCATALOG={capture_catalog|creator.DTLCCATALOG}

Optional. The name of the PowerExchange capture catalog table in the format *creator.table_name*.

Default is *creator.DTLCCATALOG*, where *creator* is the user ID that is used to connect to the database.

DBCONN=database_name

Optional. The name of the override database that you want to connect to for data extraction instead of the database that is specified for the registration group. The override database must contain tables and columns that are identical to those in the original database. The registration tag names and extraction map names include the original database name.

EPWD=encrypted_password

Optional. An encrypted password that is used with the user ID in the USERID parameter for database access.

If you specify this parameter, you must also specify either the USERID parameter. However, do not also specify the PASSWORD parameter.

Tip: You can create encrypted passwords in the PowerExchange Navigator.

LARGEOPS=number of operations

Optional. Overrides the default value that PowerExchange uses to identify transactions as large transactions for reporting purposes. Enter the number of DML operations (inserts, updates, and deletes), in thousands, that a transaction must contain to be considered a large transaction.

PowerExchange issues status messages for large transactions that meet this criteria. If PowerExchange issues too many messages, you can increase this value to limit the number of messages.

Valid values are 1 through 2147483 (1000 through 2,147,483,000 operations). The default value is one half of the MEMCACHE parameter value rounded up to the nearest thousand. Based on the default MEMCACHE value of 1024 KB, the default LARGEOPS value is 1000 (1,000,000 operations).

Note: If a committed transaction spans multiple partitions in a DB2 database, PowerExchange reports the number of SQL operations and transaction size across all of the partitions.

MEMCACHE={cache_size|1024}

Optional. The maximum memory cache size, in kilobytes, that PowerExchange can allocate to reconstruct complete UOWs.

Enter a number from 0 through 2147483647. Default is 1024 KB. If you enter 0, the memory cache size is limited only by the available memory on the system. Informatica recommends that you enter 0.

For each extraction session, PowerExchange keeps all changes for each pending UOW in the memory cache until it processes the end-UOW record. PowerExchange incrementally allocates memory cache up to the limit that this parameter specifies. If the memory cache is too small to hold all of the changes in the pending UOWs, PowerExchange spills the changes in a UOW to sequential files, called UOW spill files, on disk.

Each UOW spill file contains change data from one UOW. A UOW might require multiple UOW spill files to hold all of the changes for that UOW. If the change stream contains multiple large UOWs and the memory cache is insufficient, PowerExchange might create numerous UOW spill files.

PowerExchange processes the change stream more efficiently if it does not need to use UOW spill files. A large number of UOW spill files can degrade extraction performance and cause disk space shortages.

Important: If the change stream contains small UOWs, the default value might be sufficient. However, the default value is often too small to eliminate UOW spill files.

The location in which PowerExchange allocates the UOW spill files varies by operating system, as follows:

- For Linux and UNIX, PowerExchange uses the current directory by default. To use a different directory, you must specify the TMPDIR environment variable.

PowerExchange names the UOW spill files using the prefix "dtlq" and the operating system function tempnam.

Note: The UOW spill files are temporary files that are deleted when PowerExchange closes them. These files are not visible in the directory while they are open.

- For Windows, PowerExchange uses the current directory by default for UOW spill files. To use a different directory, specify the TMP environment variable.

PowerExchange names the UOW spill file names using the prefix "dtlq" and the Windows _tempnam function.

Warning: PowerExchange allocates the cache size for each extraction operation. If you use a large MEMCACHE value and run many concurrent extraction sessions, memory constraints can occur.

MONITORINT=minutes

Optional. The time interval, in minutes, at which PowerExchange checks transaction activity for long outstanding transactions and large transactions. A long outstanding transaction is one that remains active for two monitoring intervals, and a large transaction is one that meets the LARGEOPS criteria. When this interval elapses, PowerExchange issues messages that identify the large transactions and long outstanding transactions and report their processing activity. PowerExchange also issues

messages that identify the current position in the change stream. Valid values are 0 through 720. A value of 0 disables monitoring. Default is 5.

PASSWORD=*password*

Optional. A clear text password that is used with the user ID in the USERID parameter for database access.

If you specify this parameter, you must also specify either the USERID parameter. However, do not also specify the EPWD parameter.

RSTRADV=*seconds*

The time interval, in seconds, that PowerExchange waits before advancing restart and sequence tokens for a registered data source during periods when UOWs do not include any changes of interest for the data source. When the wait interval expires, PowerExchange returns the next committed "empty UOW," which includes only updated restart information.

Enter a number from 0 through 86400. No default is provided.

PowerExchange resets the wait interval to 0 when one of the following events occur:

- PowerExchange completes processing a UOW that includes changes of interest.
- PowerExchange returns an empty UOW because the wait interval expired without PowerExchange receiving any changes of interest.

For example, if you specify 5, PowerExchange waits 5 seconds after it completes processing the last UOW or after the previous wait interval expires. Then PowerExchange returns the next committed empty UOW that includes the updated restart information and resets the wait interval to 0.

If you do not specify RSTRADV, PowerExchange does not advance restart and sequence tokens for a registered source during periods when PowerExchange receives no changes of interest. When PowerExchange warm starts, it reads all changes, including those not of interest for CDC, from the restart point.

Attention: A value of 0 can degrade performance. In addition to the UOWs that contain changes for registered sources of interest, PowerExchange returns an empty UOW for every UOW that does not contain changes for the registered sources of interest.

SPACEPRI={primary_space|2147483647}

Optional. The amount of disk space, in bytes, that PowerExchange uses to allocate UOW spill files as temporary files.

Enter a number from 1 through 2147483647. Default is 2147483647 bytes.

THREADING={AUTO|MAX|NONE|nn}

Optional. Controls the number of threads that the UDB CAPI uses to capture change data from a DB2 database. Use this parameter to improve the performance of capture processing. If you have a partitioned database, you can use a maximum of one thread for each database partition node plus two additional threads for CAPI and merge processing.

Valid values are:

- **AUTO.** Use up to nine threads.
- **MAX.** Use one thread for each database partition plus two additional threads for CAPI and merge processing. The maximum number of threads is 99.
- **NONE.** Do not use multiple threads for capture processing.

- **nn.** A user-specified number of threads. Valid values are 1 to 99. For a partitioned database, if you enter a value that exceeds the sum of (*number_of_database_partitions* + 2), the CAPI does not use the excess threads.

Default is AUTO.

UDBSchema=*schema*

Optional. A schema name that overrides the schema name in capture registrations.

UPDINT={seconds|600}

Optional. The minimum number of seconds that PowerExchange must wait after encountering a virtual time stamp (VTS) in the DB2 log records for a partition before writing a positioning entry to the PowerExchange capture catalog table. The positioning entry, which contains a log sequence number (LSN) and VTS, indicates the location in the DB2 logs.

Enter a number from 1 through 2147483647. Default is 600 seconds.

Note: The minimum number of records that is specified in the UPDREC parameter must also be met before PowerExchange can write positioning entries to the capture catalog table.

UPDREC={records|1000}

Optional. The minimum number of DB2 log records that PowerExchange must read for a partition before writing a positioning entry to the PowerExchange capture catalog table. The positioning entry contains an LSN and VTS and indicates a location in the DB2 logs.

Enter a number from 1 through 2147483647. Default is 1000 records.

Note: The minimum wait period that is specified in the UPDINT parameter must also be met before PowerExchange can write positioning entries to the capture catalog table.

USERID=*user_id*

Optional. A database user ID that has SYSADM or DBADM authority.

If you specify this parameter, you must also specify the PASSWORD or EPWD parameter.

CAPI_CONNECTION - UOWC Statement

The UOWC CAPI_CONNECTION statement specifies a named set of parameters that the Consumer API (CAPI) uses for the UOW Cleanser.

In the change stream for some data sources, changes from multiple UOWs are intermingled. The UOW Cleanser reconstructs the intermingled changes read from the change stream into complete UOWs in chronological order based on end time.

Operating Systems: i5/OS, Linux, UNIX, Windows, and z/OS

Data Sources: Db2 for i5/OS and z/OS CDC sources

Related Statements: CAPI_CONNECTION - AS4J and CAPI_CONNECTION - LRAP

Required: Yes, for CDC for the specified sources

Syntax:

```
CAPI_CONNECTION= ([DLLTRACE=trace_id]
                  ,NAME=capi_connection_name
                  [,TRACE=trace_name]
                  ,TYPE=(UOWC
                        ,CAPINAME=source_capi_name
                        [,BLKSIZE=block_size])
```

```

[,CUOWS={number_of_concurrent_UOWs|34}]
[,DATACLASS=data_class]
[,LARGEOPS=number_of_operations]
[,MEMCACHE={cache_size|1024}]
[,MONITORINT={minutes|5}]
[,RSTRADV=seconds]
[,SPACEPRI={primary_space|50}]
[,SPACETYP={BLK|TRK|CYL}]
[,SPILLKEEP=number_of_spill_files]
[,STORCLASS=storage_class]
[,TIMESTAMP={LOG|COMMIT}]
[,UNIT=unit]
)
)

```

Parameters:

DLLTRACE=trace_id

Optional. A user-defined name for the TRACE statement that activates internal DLL tracing for this CAPI.

Specify this parameter only at the direction of Informatica Global Customer Support.

NAME=capi_connection_name

Required. A unique user-defined name for this CAPI_CONNECTION statement.

Maximum length is eight alphanumeric characters.

TRACE=trace_name

Optional. A user-defined name for the TRACE statement that activates the common CAPI tracing.

Specify this parameter only at the direction of Informatica Global Customer Support.

TYPE=(UOWC, ...)

Required. The type of CAPI_CONNECTION statement. For the UOW Cleanser, this value must be UOWC.

CAPINAME=capi_name

Required. The value of the NAME parameter in the related source-specific CAPI_CONNECTION statement, which can be one of the following statement types:

- AS4J CAPI_CONNECTION statement for Db2 for i (i5/OS) sources
- LRAP CAPI_CONNECTION statement for data sources on z/OS

BLKSIZE=block_size

Optional. The block size, in bytes, for the sequential UOW spill files that the UOW Cleanser creates when the memory cache cannot hold all changes for a UOW.

The following table shows valid values by CDC source type:

Data Source Type	Valid Values	Default Value
Db2 for i (i5/OS)	A number from 8 through 32760	32760
z/OS data sources	A number from 8 through 32760	18452

CUOWS=number_of_concurrent_UOWs

The expected number of concurrent UOWs that contain data of CDC interest. If you specified a large MEMCACHE value, have excessive spill file allocations, and notice a large number of concurrent

UOWs reported in message PWX-10742 or PWX-10782, increase this parameter value to cause less memory cache to be used per UOW. By reducing memory cache use per UOW, more concurrent UOWs can be accommodated without allocating spill files. This parameter in no way restricts the maximum number of concurrent UOWs that the UOW Cleanser can process.

Valid values are 1 through 65535. Default is 34.

DATACLASS=*data_class*

Optional. On z/OS, the SMS data class that the UOW Cleanser uses when allocating the sequential UOW spill files. If you do not specify this parameter, the SMS ACS routines can assign the data class.

LARGEOPS=*number of operations*

Optional. Overrides the default value that PowerExchange uses to identify transactions as large transactions for reporting purposes. Enter the number of DML operations (inserts, updates, and deletes), in thousands, that a transaction must contain to be considered a large transaction.

PowerExchange issues status messages for large transactions that meet this criteria. If PowerExchange issues too many messages, you can increase this value to limit the number of messages.

Valid values are 1 through 2147483 (1000 through 2,147,483,000 operations). The default value is one half of the MEMCACHE parameter value rounded up to the nearest thousand. Based on the default MEMCACHE value of 1024 KB, the default LARGEOPS value is 1000 (1,000,000 operations).

MEMCACHE={*cache_size*|1024}

Optional. The maximum memory cache size, in kilobytes, that PowerExchange allocates to reconstruct complete UOWs.

Enter a number from 0 through 2147483647. Default is 1024 KB. If you enter 0, the memory cache size is limited only by the available memory on the system.

For each extraction session, PowerExchange keeps all changes for each UOW in the memory cache until it processes the end-UOW record. PowerExchange incrementally allocates memory cache up to the limit that this parameter specifies. If the memory cache is too small to hold all of the changes in a UOW, PowerExchange spills the changes to a sequential files on disk, called UOW spill files.

Each UOW spill file contains one UOW. A UOW might require multiple UOW spill files to hold all of the changes for that UOW. If the change stream contains multiple large UOWs and the memory cache is insufficient, PowerExchange might create numerous UOW spill files.

PowerExchange processes the change stream more efficiently if it does not need to use UOW spill files. A large number of UOW spill files can degrade extraction performance and cause disk space shortages.

Important: If the change stream contains small UOWs, the default value might be sufficient. However, Informatica recommends that you specify a larger value because the default value is often too small.

The location in which PowerExchange allocates the UOW spill files varies by operating system, as follows:

- For i5/OS, PowerExchange uses CRTPF command to create a physical file for UOW spill files. PowerExchange names the UOW spill files using the C/C++ `tmpnam()` function.
- For Linux and UNIX, PowerExchange uses the current directory by default for UOW spill files. To use a different directory, specify the TMPDIR environment variable.

PowerExchange names the UOW spill file names using the prefix "dtlq" and the operating system function tempnam.

Note: The UOW spill files are temporary files that are deleted when PowerExchange closes them. These files are not visible in the directory while they are open.

- For Windows, PowerExchange uses the current directory by default for UOW spill files. To use a different directory, specify the TMP environment variable.

PowerExchange names the UOW spill file using the prefix "dtlq" and the Windows _tempnam function.

- For z/OS, PowerExchange uses dynamic allocation to allocate temporary data sets for the UOW spill files. Generally, SMS controls the location of temporary data sets. If you do not use SMS to control temporary data sets, the UNIT parameter controls the location for the UOW spill files.

Because PowerExchange allocates temporary data sets for the UOW spill files, z/OS assigns these files system-generated data set names, which begin with `SYSyddd.Thhmmss.RA000.jobname`.

Warning: PowerExchange allocates the cache size for each extraction operation. If you use a large MEMCACHE value and run many concurrent extraction sessions, memory constraints can occur.

MONITORINT=minutes

Optional. The time interval, in minutes, at which PowerExchange checks transaction activity for long outstanding transactions and large transactions. A long outstanding transaction is one that remains active for two monitoring intervals, and a large transaction is one that meets the LARGEOPS criteria. When this interval elapses, PowerExchange issues messages that identify the large transactions and long outstanding transactions and report their processing activity. PowerExchange also issues messages that identify the current position in the change stream. Valid values are 0 through 720. A value of 0 disables monitoring. Default is 5.

RSTRADV=seconds

The time interval, in seconds, that PowerExchange waits before advancing restart and sequence tokens for a registered data source during periods when UOWs do not include any changes of interest for the data source. When the wait interval expires, PowerExchange returns the next committed "empty UOW," which includes only updated restart information.

Enter a number from 0 through 86400. No default is provided.

PowerExchange resets the wait interval to 0 when one of the following events occur:

- PowerExchange completes processing a UOW that includes changes of interest.
- PowerExchange returns an empty UOW because the wait interval expired without PowerExchange receiving any changes of interest.

For example, if you specify 5, PowerExchange waits five seconds after it completes processing the last UOW or after the previous wait interval expires. Then PowerExchange returns the next committed empty UOW that includes the updated restart information and resets the wait interval to 0.

If you do not specify RSTRADV, PowerExchange does not advance restart and sequence tokens for a registered source during periods when PowerExchange receives no changes of interest. When PowerExchange warm starts, it reads all changes, including those not of interest for CDC, from the restart point.

For Db2 for i (i5/OS) sources, Informatica recommends that you use this parameter if the change records that PowerExchange reads from i5/OS journal receivers are created under commitment

control. If the change records are created without commitment control, do not specify this parameter.

Attention: A value of 0 can degrade performance. In addition to the UOWs that contain changes for registered sources of interest, PowerExchange returns an empty UOW for every UOW that does not contain changes for the registered sources of interest.

SPACEPRI={primary_space|50}

Optional. On z/OS, the amount of primary space that the UOW Cleanser uses for allocating UOW spill files. The SPACETYP parameter indicates the type of space units.

Enter a number from 1 through 16777215. Default is 50 blocks.

The UOW Cleanser does not use secondary space. Instead, when a spill file becomes full, the UOW Cleanser allocates another spill file of the same size.

SMS ACS routines can override the UOW spill file size.

Note: On i5/OS, the UOW Cleanser allocates UOW spill files as physical files with SIZE(*NOMAX), which means that the maximum spill file size is controlled by the system maximum file size. On Linux, UNIX, and Windows, PowerExchange allocates UOW spill files as temporary files that are 2 GB in size.

SPACETYP={BLK|TRK|CYL}

Optional. On z/OS, the type of units in which the primary space for UOW Cleanser allocation of UOW spill files is expressed.

Options are:

- **BLK.** Blocks.
- **CYL.** Cylinders.
- **TRK.** Tracks.

Default is BLK.

SPILLKEEP=*number_of_spill_files*

Optional. The number of spill files that the UOW Cleanser retains for re-assignment. The UOW Cleanser retains spill files instead of deallocating them so that the files are available to be reassigned to new transactions. This feature is intended to prevent excessive file deallocation and allocation activity.

Valid values are 0 through 999. On z/OS and i5/OS, the default is 3. On Linux, UNIX, and Windows, the default is 0.

STORCLASS=*storage_class*

Optional. On z/OS, the SMS storage class name that the UOW Cleanser uses to allocate UOW spill files.

TIMESTAMP={LOG|COMMIT}

Optional. The type of timestamp that PowerExchange records in the generated DTL__CAPXTIMESTAMP column of each change record for a transaction. Specify this parameter only if you want to display the commit timestamp instead of the timestamp from the source logs or data sets.

Options are:

- **LOG.** The timestamp of a change on the source database, as recorded by the DBMS in the source database logs or data sets near the time when the change is made. For more information, see [Appendix A, “DTL__CAPXTIMESTAMP Time Stamps” on page 425](#).
- **COMMIT.** The timestamp of the transaction commit on the source database. Specify this option if you use the timestamp to calculate latency.

Default is LOG.

UNIT=unit

Optional. On z/OS, the generic or esoteric unit name that the UOW Cleanser uses to allocate UOW spill files.

CAPI_SRC_DFLT Statement

The CAPI_SRC_DFLT statement specifies the CAPI_CONNECTION statement that PowerExchange uses by default for a specific data source type when a CAPI connection override is not supplied.

Informatica recommends that you specify this statement for each source type to ensure that the appropriate type of CAPI_CONNECTION statement is used for accessing a source. If you have multiple source types that run on different platforms and do not specify a CAPI_SRC_DFLT statement for each source type, source access problems might occur in certain situations during PowerCenter data previews or CDC sessions.

Operating Systems: All

Data Sources: All

Related Statements: CAPI_CONN_NAME and CAPI_CONNECTION

Required: No

Syntax:

```
CAPI_SRC_DFLT=(source_type  
               , capi_connection_name)
```

Parameters:

source_type

Required. The CDC source type. The following table identifies the valid source-type options:

Option	Source Type
ADA	Adabas sources
AS4	Db2 for i (i5/OS) sources
CAPX	Sources for which you are extracting data in continuous extraction mode and using the PowerExchange Logger for Linux, UNIX, or Windows or PowerExchange Condense
DB2	Db2 for z/OS sources
DCM	CA Datacom sources
IDL	CA IDMS/DB log-based CDC sources

Option	Source Type
IML	IMS log-based CDC sources
IMS	IMS synchronous CDC sources
MSS	Microsoft SQL Server sources
MYS	MySQL sources
ORA	Oracle sources
PGS	PostgreSQL sources
UDB	Db2 sources on Linux, UNIX, or Windows
VSAM or VSM	VSAM sources

capi_connection_name

Required. Unique name of the CAPI_CONNECTION statement to use as the default for the specified source type. This name must match the NAME value in a CAPI_CONNECTION statement that has a TYPE value that is compatible with the CAPI_SRC_DFLT *source_type*.

The following table shows, for each CAPI_SRC_DFLT option, the compatible CAPI_CONNECTION type:

CAPI_SRC_DFLT Option	CAPI_CONNECTION Statement Type
AS4	UOWC
CAPX	CAPX
ADA, DB2, DCM, IDL, IDM, IML, IMS, VSAM, or VSM	UOWC
ORA	ORAD for PowerExchange Express CDC for Oracle
MSS	MSQL
MYS	MYSQL
PGS	PG
UDB	UDB

Usage Notes:

- If you define multiple CAPI_CONNECTION statements for a source type, you can define a CAPI_SRC_DFLT statement to identify the default CAPI_CONNECTION for that source type. The CAPI_SRC_DFLT statement must point to a CAPI_CONNECTION statement of a compatible type.
- You can optionally define a CAPI_CONN_NAME statement that specifies an overall default statement, out of all of the CAPI_CONNECTION statements in the DBMOVER file.

- Instead of or in addition to specifying defaults, you can use the following CAPI connection name overrides to point to a specific CAPI_CONNECTION statement for CDC sessions or database row tests:
 - For CDC sessions, use the **CAPI Connection Name Override** attribute on the PWX CDC application connection.
 - For PowerExchange Condense, use the CONN_OVR parameter in CAPTPARM configuration file.
 - For the PowerExchange Logger for Linux, UNIX, and Windows, use the CONN_OVR parameter in pwxcl.cfg configuration file.
 - For DTLUAPPL utility operations that generate restart tokens, use the CONN_OVR parameter in the DTLUAPPL control statement.
 - For CAPXRT database row tests in the PowerExchange Navigator, use the **CAPI Connection Name** value in the **CAPXRT Advanced Parameters** dialog box. If you add an SQL statement for generating restart tokens, you can include the CONNAME parameter to point to the override CAPI_CONNECTION.
 - For PowerExchange ODBC connections, use the DTLCONN_OVR parameter in the odbc.ini file or the SQL escape sequence override DTLCONN_OVR.

RELATED TOPICS:

- [“Consumer API \(CAPI\) Connection Statements” on page 204](#)

CAPT_PATH Statement

The CAPT_PATH statement specifies the path to a directory on a Linux, UNIX, or Windows system that contains the control files for CDC.

Operating Systems: Linux, UNIX, and Windows

Data Sources: Db2, Microsoft SQL Server, MySQL, Oracle, and PostgreSQL sources on Linux, UNIX, or Windows, and if you use remote logging, data sources on IBM i (i5/OS) or z/OS

Related Statements: CAPT_XTRA

Required: Yes for CDC sources on Linux, UNIX, and Windows

Syntax:

```
CAPT_PATH=path
```

Value: For the *path* variable, enter the path to the local directory that contains the following control files for CDC:

- CCT file, which contains capture registrations.
- CDEP file, which contains application names for any PowerCenter extractions that use ODBC connections.
- CDCT file, which contains information about PowerExchange Logger for Linux, UNIX, and Windows log files.

This directory can be one that you created specifically for these files or another directory.

Informatica recommends that you use a unique directory name to separate these CDC objects from the PowerExchange code. This practice makes migrating to a another PowerExchange version easier.

Default is the PowerExchange installation directory.

Usage Notes:

- PowerExchange C-ISAM control files, such as the CCT, CDEP, and CDCT files, must be stored on local disk. Do not locate these files in SAN or NAS storage.

- To provide a path to the directory that contains extraction maps, use the CAPT_XTRA statement.

CAPT_XTRA Statement

The CAPT_XTRA statement specifies the path to the local directory on a Linux, UNIX, or Windows system that stores extraction maps for CDC.

Operating Systems: Linux, UNIX, and Windows

Data Sources: Db2, Microsoft SQL Server, MySQL, Oracle, and PostgreSQL sources on Linux, UNIX, or Windows

Related Statements: CAPT_PATH

Required: Yes for CDC sources on Linux, UNIX, and Windows

Syntax:

```
CAPT_XTRA=path
```

Value: For the *path* variable, enter the path to the local directory that stores extraction maps.

This directory can be a directory that you created specifically for extraction maps or another directory. Default is the PowerExchange installation directory.

Usage Notes: To specify the directory that contains the CCT and CDEP control files for CDC, use the CAPT_PATH statement.

CMDNODE Statement

The CMDNODE statement specifies connection information for a PowerExchange process that is the target of pwxcmd commands.

Include the CMDNODE statement in the dbmover.cfg file on the Linux, UNIX, or Windows system from which you issue pwxcmd commands.

Operating Systems: Linux, UNIX, and Windows

Related Statements: SVCNODE

Required: No

Syntax:

```
CMDNODE=(service_name
        ,{CONDENSE|ECCR|LISTENER}
        ,host_name
        ,connect_port
)
```

Parameters:

service_name

Required. User-defined name of the command-handling service for the PowerExchange process to which you want to send pwxcmd commands. Use this service name when issuing pwxcmd commands to this PowerExchange process. For connection to a PowerExchange log-based ECCR or Datacom table-based ECCR on z/OS, this value is the ECCR name. Ensure that this value matches the ECCR name specified in the corresponding SVCNODE statement.

{CONDENSE|ECCR|LISTENER}

Required. The PowerExchange service type.

Options are:

- **CONDENSE.** PowerExchange Condense or PowerExchange Logger for Linux, UNIX, and Windows.
- **ECCR.** A PowerExchange Adabas, IDMS, or IMS log-based ECCR or Datacom table-based ECCR on z/OS.
- **LISTENER.** A PowerExchange Listener.

No default value is provided.

host_name

Required. The host name or IP address of the target system to which you want to send commands.

connect_port

Required. The port number on which the command-handling service for the PowerExchange process listens for pwxcmd commands. This port number must match the port number in the corresponding SVCNODE statement.

CODEPAGE Statement

The CODEPAGE statement specifies the code pages that PowerExchange uses for operating system information, data, and literal values in SQL statements.

Operating Systems: All

Related Statements: ADABASCODEPAGE, CONSOLE_CODEPAGE, DB2_BIN_CODEPAGE, DB2CODEPAGE, DTLMSG_CODEPAGE, LOG_CODEPAGE, and ORACLECODEPAGE

Required: No

Syntax:

```
CODEPAGE=(ctrl_cp
           ,data_cp
           ,SQL_cp)
```

Parameters:

ctrl_cp

Required. Control code page identifier for operating system objects, such as program names and file names.

data_cp

Required. Code page identifier that overrides the code page of data. For example, when you extract data on Windows from z/OS sources, this value overrides the z/OS code page for that data.

SQL_cp

Required. Code page identifier for literal values used in SQL statements.

Usage Notes:

- If you do not define the CODEPAGE statement, PowerExchange uses the default code page for the operating system for all parameters. The following table lists the default values by operating system:

Operating System	Default Code Page Values
i5/OS	IBM-037,IBM-037,IBM-037
Linux, UNIX, and Windows	ISO-8859,ISO-8859,ISO-8859
z/OS	IBM-037,IBM-037,IBM-037

- This statement does not apply to the PowerExchange Navigator. The PowerExchange Navigator always uses the UTF-8 code page.
- If you click **Advanced Parms** on the **General Parameters** page of the z/OS Installation Assistant, you can define the CODEPAGE_CONTROL, CODEPAGE_DATA, and CODEPAGE_SQL parameters. These values are used to populate the CODEPAGE statement in the DBMOVER file.
 - The CODEPAGE_CONTROL parameter corresponds to the *ctrl_cp* parameter in the CODEPAGE statement.
 - The CODEPAGE_DATA parameter corresponds to the *data_cp* parameter in the CODEPAGE statement.
 - The CODEPAGE_SQL parameter corresponds to the *SQL_cp* parameter in the CODEPAGE statement.

RELATED TOPICS:

- [“PowerExchange Globalization” on page 346](#)

COLON Statement

The COLON statement defines the character that PowerExchange uses as a the delimiter character for time values in times and time stamp fields.

Operating Systems: All

Required: No

Syntax:

```
COLON={delimiter|:}
```

Value: For the *delimiter* variable, enter any character. Default is the colon (:) character.

For example, if you accept the default value, PowerExchange writes time fields in the following format:

```
HH:MM:SS
```

COMPRESS Statement

The COMPRESS statement controls whether PowerExchange uses its proprietary compression routines to compress data that it transmits across the network.

Operating Systems: All

Required: No

Syntax:

```
COMPRESS={N|Y}
```

Valid Values:

- **N.** The PowerExchange Listener does not compress data.
- **Y.** The PowerExchange Listener compresses data that it transmits across the network on the source system, and decompresses the data on the target system.

Default is N.

CONSOLE_CODEPAGE Statement

The CONSOLE_CODEPAGE statement specifies the code page that PowerExchange uses to display message text on consoles.

Operating Systems: Linux, UNIX, and Windows

Related Statements: CODEPAGE

Required: No

Syntax:

```
CONSOLE_CODEPAGE=code_page
```

Value: For the *code_page* variable, enter a code page identifier. For example, to specify the console code page where the operating system locale uses Japanese characters, define the following statement:

```
CONSOLE_CODEPAGE=CP943
```

Default is the control code page. On Linux and UNIX, the console emulator assumes the data to be in code page UTF-8. On Windows, the operating system uses the OS locale derived from the user internationalization settings.

CONSOLE_MSG Statement

The CONSOLE_MSG statement specifies whether to write certain messages to the domain log in addition to writing them to the PowerExchange log file on the Integration Service machine. Many of these messages are related to errors that the PowerExchange Listener returns.

Operating Systems: Linux, UNIX, and Windows

Required: No

```
CONSOLE_MSG={N | Y}
```

Valid Values:

- **N.** PowerExchange does not write the messages to the domain log.
- **Y.** PowerExchange writes the messages to the domain log.

Default is Y.

Usage Notes:

- In certain cases excessive messages in a session might cause node failure with a Java heap space out-of-memory condition. If this situation occurs, define CONSOLE_MSG=N.
- When CONSOLE_MSG=N, PowerExchange messages are not shown in a console session, that is, on a Linux or UNIX terminal or in a Windows DOS session. The messages are still written to the PowerExchange log file.

CONSOLE_TRACE Statement

The CONSOLE_TRACE statement controls whether PowerExchange writes messages from TRACE statements to the DTLOUT data set for z/OS or to the QPRINT output for i5/OS, in addition to writing these messages to the PowerExchange message log file.

Operating Systems: i5/OS and z/OS

Required: No

Syntax:

```
CONSOLE_TRACE={N|Y}
```

Valid Values:

- **N.** PowerExchange writes messages from TRACE statements to the PowerExchange message log file.
- **Y.** PowerExchange writes messages from TRACE statements to the PowerExchange message log file. On z/OS, PowerExchange also writes these messages to the DTLOUT (redirected stdout) data set. On i5/OS, PowerExchange also writes the messages to the QPRINT output for the PowerExchange Listener.

Default is N.

CONVCHAR Statement

The CONVCHAR statement controls how PowerExchange handles conversion errors for CHAR and VARCHAR columns that it detects during bulk data movement read operations.

Data Sources: All

Required: No

Syntax:

```
CONVCHAR=(replacement_character  
          [, {replacement_option|NULL}]  
          [, maximum_length]  
          [, {ERROR|NOERROR}]  
          )
```

Parameters:

replacement_character

Required. The character that PowerExchange uses to replace the data of CHAR and VARCHAR columns that have conversion errors.

{*replacement_option*|NULL}

Optional. Controls what PowerExchange uses to replace the data of nullable columns that have conversion errors.

Enter one of the following options:

- **NULL.** By default, PowerExchange replaces the data of nullable columns with nulls.
- **REPLACE.** PowerExchange replaces the content of nullable columns with the replacement character.

Default is NULL.

maximum_length

Optional. Maximum number of replacement characters that PowerExchange uses to replace the content of columns that have conversion errors.

{ERROR|NOERROR}

Optional. Controls whether PowerExchange treats columns that have conversion errors as fatal.

Enter one of the following options:

- **ERROR.** By default, PowerExchange treats a conversion error for a column as fatal and terminates the data movement operation.

PowerExchange writes the PWX-000144 error message to the PowerExchange message log, which includes the column name and the row number where PowerExchange detected the conversion error.

- **NOERROR.** PowerExchange replaces the data in the column with the conversion error with the replace character and continues.

PowerExchange writes PWX-00142 message to the PowerExchange message log, which includes the number of conversion errors detected.

Default is ERROR.

CPX_DIR Statement

The CPX_DIR statement specifies the name of the library on i5/OS in which PowerExchange stores extraction maps for DB2 for i5/OS CDC sources.

Operating Systems: i5/OS

Data Sources: DB2 for i5/OS CDC

Required: No

Syntax:

```
CPX_DIR=library
```

Value: For the *library* variable, enter the name of the i5/OS library in which PowerExchange stores extraction maps for DB2 for i5/OS CDC sources. Default is CPXLIB.

Usage Notes: When you run the CRTPWXENV command during PowerExchange installation, PowerExchange creates this library and includes its name in the DBMOVER configuration member. The name is based on the CPXLIB parameter on the CRTPWXENV command.

CREDENTIALS_CASE Statement

The CREDENTIALS_CASE statement controls the case that PowerExchange uses for operating system user IDs and passwords.

Operating Systems: All

Required: No

Syntax:

```
CREDENTIALS_CASE={A|D|S}
```

Valid Values:

- **A.** On z/OS or i5/OS, PowerExchange processes and passes user IDs and passwords to the operating system for authentication, as follows:

1. PowerExchange converts the user ID to uppercase.

2. PowerExchange checks whether the operating system is configured to handle mixed-case passwords.

- If so, PowerExchange passes the user ID in uppercase and the password in the case that you supplied it to the operating system for authentication.
- If not, PowerExchange converts the password to uppercase and passes the user ID and password to the operating system for authentication.

On Linux, UNIX, or Windows, PowerExchange passes the user ID and password in the case that you supplied them to the operating system for authentication.

- **D.** On i5/OS or z/OS, PowerExchange converts user IDs and passwords to uppercase and then passes them to the operating system for authentication.

On Linux, UNIX, or Windows, PowerExchange passes the user ID and password in the case that you supplied them to the operating system for authentication.

- **S.** On i5/OS or z/OS, PowerExchange converts the user ID to uppercase and leaves the password in the case that you supplied it. Then, PowerExchange passes the user ID and password to the operating system for authentication.

On Linux, UNIX, or Windows, PowerExchange passes the user ID and password in the case that you supplied them to the operating system for authentication.

Default is D.

DATAMAP_SERVER Statement

The DATAMAP_SERVER statement specifies the node name of the PowerExchange Listener that acts as the server for access requests to the file that stores data maps.

Use this statement to reduce overhead in an environment where multiple PowerExchange Listeners are running and make frequent open and close requests on the data map file.

Operating Systems: z/OS

Related Statements: DM_SUBTASK and NODE

Required: No

Syntax:

```
DATAMAP_SERVER=node_name
```

Value: For the *node_name* variable, enter the node name from a NODE statement in the DBMOVER member that points to the PowerExchange Listener that accesses the DATAMAPS data set.

Generally, you designate one PowerExchange Listener as the data map server. To do so, define the DATAMAP_SERVER statement in the DBMOVER members of the PowerExchange Listeners that contact the data map server.

Usage Notes:

- If you have two or more PowerExchange Listeners that share the same DATAMAPS data set, you can configure PowerExchange to use the PowerExchange Listener that starts first as the data map server. If you have three or more PowerExchange Listeners that share the same DATAMAPS data set, you must designate one of them as the data map server.

To use a PowerExchange Listener as the data map server, configure the following statements in the DBMOVER member for each PowerExchange Listener:

- In the DATAMAP_SERVER statement, specify the node name of the other PowerExchange Listener.

- Specify Y for the DM_SUBTASK statement.

The first PowerExchange Listener that starts becomes the data map server and the other PowerExchange Listeners access data maps through it. PowerExchange writes the following messages to the PowerExchange message log file:

- The PWX-02804 message to the PowerExchange message log file of the PowerExchange Listener that acts as the data map server.
- The PWX-02800 and PWX-02805 messages to the PowerExchange message log files of the PowerExchange Listeners that do not act as the data map server. The PWX-02805 message indicates node name, IP address, and port number of the PowerExchange Listener that is the data map server.
- If you also specify DM_SUBTASK=Y and a PowerExchange Listener holds an exclusive lock on the DATAMAPS data set, enter DATAMAP_SERVER=*node_name* to enable other tasks, such as the IMS synchronous ECCR or a netport job, to access the data set. Otherwise, the tasks fail.

Alternatively, use data maps caching. Informatica recommends this approach to improve performance and to avoid access problems that can occur if the PowerExchange Listener data maps server stops. To implement data maps caching, specify DM_SUBTASK=R and DMXCACHE_MAX_MEMORY_MB=20. With these settings, the PowerExchange Listener opens the data set in read-only mode and allows other tasks to access the data set.

DATERANGE Statement

The DATERANGE statement specifies the range of years that is used to validate data in DATE and TIMESTAMP fields.

Operating Systems: All

Required: No

Syntax:

```
DATERANGE=({start_year|1800}
            ,{end_year|2200})
```

Parameters:

{start_year|1800}

Required. The year that PowerExchange uses as the starting year in the date range for validating DATE and TIMESTAMP fields. Default is 1800.

{end_year|2200}

Required. The year that PowerExchange uses as the ending year in the date range for validating DATE and TIMESTAMP fields. This value must be equal to or greater than the value specified for the starting year. Default is 2200.

Usage Notes: PowerExchange treats data values outside the range specified in the DATERANGE statement as not valid. On the **Data Checking** tab in the PowerExchange Navigator, you can select the action that PowerExchange performs when it detects a value in a date field that is not valid. By default, PowerExchange writes the PWX-02152 and PWX-02153 messages to the PowerExchange message log and stops the data extraction operation.

DB2_BIN_AS_CHAR Statement

The DB2_BIN_AS_CHAR statement controls whether PowerExchange considers data in CHAR and VARCHAR columns that you define with the FOR BIT DATA clause as character data.

To override character data that is incorrectly assigned with CCSID 65535, use this statement in conjunction with the DB2_BIN_CODEPAGE statement.

Operating Systems: i5/OS

Data Sources: DB2 for i5/OS

Related Statements: DB2_BIN_CODEPAGE

Required: No

Syntax:

```
DB2_BIN_AS_CHAR={N|Y}
```

Valid Values:

- **N.** Data in CHAR and VARCHAR columns defined with the FOR BIT DATA clause is binary data.
PowerExchange does not perform code page translation for binary data.
- **Y.** Data in CHAR and VARCHAR columns defined with the FOR BIT DATA clause is character data.

Default is N.

DB2_BIN_CODEPAGE Statement

The DB2_BIN_CODEPAGE statement defines the single-byte and multibyte CCSID values that PowerExchange uses to process character data in columns that you define with the FOR BIT DATA clause, if you specify Y for the DB2_BIN_AS_CHAR statement.

To override character data incorrectly assigned with CCSID 65535, use this statement in conjunction with the DB2_BIN_AS_CHAR statement.

Operating Systems: i5/OS

Data Sources: DB2 for i5/OS

Related Statements: DB2_BIN_AS_CHAR

Required: No

Syntax:

```
DB2_BIN_CODEPAGE=(sbcscsid  
                  ,dbcsccsid  
)
```

Parameters:

sbcscsid

Required. CCSID value for single-byte data. Valid values are from 1 through 65534.

dbcsccsid

Required. CCSID value for multibyte data. Valid values are from 1 through 65534.

Example: The following statement defines CCSID values for Japanese data:

```
DB2_BIN_CODEPAGE=(8482,1390)
```

DB2_ERRORFILE Statement

The DB2_ERRORFILE statement specifies the name of the user-customized SQL error file that PowerExchange uses for DB2 bulk data movement operations.

Operating Systems: All

Data Sources: DB2

Required: No

Syntax:

```
DB2_ERRORFILE=file_name
```

Value: For the *file_name* variable, enter the name of the file or member that contains the SQL error codes that you want PowerExchange to treat as either recoverable or fatal.

The content of *file_name* varies based on the operating system on which DB2 runs, as follows:

- **i5/OS.** Specify the library and file, and optionally member.
- **z/OS.** Specify the sequential or PDS and member.
- **Linux, UNIX, and Windows.** Specify the path and file name.

On i5/OS or z/OS, if you include a file or member name, enclose the file or member name in double quotation marks ("").

Usages Notes: PowerExchange supplies sample error files.

The following table lists the sample error files for each DB2 data source:

Data Source	Sample Error File
DB2 for z/OS	DB2ERR member in RUNLIB
DB2 for i5/OS	DB2ERR member in <i>datalib</i> /CFG
DB2 for Linux, UNIX, and Windows	db2err.act in the PowerExchange installation directory

DB2CODEPAGE Statement

The DB2CODEPAGE statement defines the single-byte, graphic, and mixed CCSIDs that PowerExchange uses to process CHAR and VARCHAR column data in a DB2 for z/OS subsystem that is involved in bulk data movement.

Operating Systems: z/OS

Data Sources: DB2 for z/OS

Required: No

Syntax:

```
DB2CODEPAGE=(db2_subsystem
  [,DB2TRANS={P|N|R}]
  [,MIXED={N|Y}]
  [,EBCDIC_CCSID=({sbcsc_ccsid|037}
    ,{graphic_ccsid|037}
    ,{mixed_ccsid|037})]
  [,ASCII_CCSID=({sbcsc_ccsid|850}
    ,{graphic_ccsid|65534}
    ,{mixed_ccsid|65534})])
```

```
[,UNICODE_CCSID=({sbcsc_ccsid|367}
                  ,{graphic_ccsid|1200}
                  ,{mixed_ccsid|1208})]
[,PLAN_CCSID=({sbcsc_ccsid|037}
              ,{graphic_ccsid|037}
              ,{mixed_ccsid|037})]
[,REMAPn=(current_data_ccsid),(remapped_data_ccsid)

)
```

Parameters:

db2_subsystem

Required. The DB2 subsystem identifier (SSID) of a source or target subsystem.

DB2TRANS={P|N|R}

Indicates whether DB2 translates the encoding of data that it passes to or receives from PowerExchange. Options are:

- **P.** DB2 translates the code pages in which column data is stored into the code pages defined in the DB2 plan that was bound for PowerExchange. You must also specify the EBCDIC_CCSID parameter and optionally the PLAN_CCSID parameter. If you specify both, the PLAN_CCSID parameter takes precedence. If you have ASCII and Unicode data, you can also specify the ASCII_CCSID and UNICODE_CCSID parameters to map to the EBCDIC code pages.
Note: To use any of the *_CCSID parameters, you must set DB2TRANS to P.
- **N.** DB2 does not translate the code pages of the column data to equivalent EBCDIC code pages. PowerExchange uses the native code page in which the data is stored. You do not need to define the EBCDIC_CCSID, ASCII_CCSID, UNICODE_CCSID, or PLAN_CCSID parameters.
- **R.** DB2 translates certain user-specified data code pages to other code pages, as defined in one or more REMAPn parameters. In each REMAPn parameter, the first positional parameter identifies a data code page to remap, and the second positional parameter identifies the code page to use. Use a code page other than the code page in which the PowerExchange DB2 plan is bound.

Default is P.

MIXED={N|Y}

Indicates whether DB2 columns contain ASCII and EBCDIC character strings with mixed data and graphic data. Mixed data consists of both single-byte and double-byte characters, and graphic data consists of double-byte characters.

Options are:

- **N.** Columns with ASCII and EBCDIC character strings contain only single-byte (SBCS) data. Mixed and graphic data does not occur.
- **Y.** Columns with ASCII and EBCDIC character strings contain mixed data and graphic data.

Default is N.

EBCDIC_CCSID=({sbcsc_ccsid|037},{graphic_ccsid|037},{mixed_ccsid|037})

Required if DB2TRANS=P (the default). The CCSIDs to use for EBCDIC single-byte, graphic double-byte, and mixed (single-byte and double-byte) data.

Valid values are from 1 through 65534 for each positional subparameter. A value of 65534 indicates no CCSID.

The default values of 037,037,037 are used if you do not specify the DB2CODEPAGE statement or if you accept the values from the z/OS Installation Assistant without modification. If you changed these values

during installation, the Installation Assistant customizes the DB2CODEPAGE statement in the DBMOVER file with the values you entered.

If an EBCDIC code page does not have an ICU conversion table, or if the EBCDIC code page is different from the one in which the DB2 plan for PowerExchange is bound, you can use the PLAN_CCSID parameter to map to another EBCDIC code page that does have an ICU conversion table or that reflects the DB2 plan.

Note: DB2 delivers data to PowerExchange in an EBCDIC code page even if the data is physically stored in an ASCII or Unicode code page. The DB2 catalog tables are stored in UTF-8, which corresponds to CCSID 1208. This CCSID is remapped from the UNICODE_CCSID *mixed_ccsid* code page value to the EBCDIC_CCSID *mixed_ccsid* code page value.

On single-byte systems, either repeat the SBCS code page three times or use 65534 with the MIXED=N parameter. For example:

```
EBCDIC_CCSID=(1047,1047,1047)
```

or

```
DB2CODEPAGE=(D91G,EBCDIC_CCSID=(1047,65534,65534),MIXED=N)
```

PowerExchange then uses the single-byte EBCDIC code page.

ASCII_CCSID=({*sbcsc_ccsid*|850},{*graphic_ccsid*|65534},{*mixed_ccsid*|65534})

Optional. The CCSIDs to use for ASCII single-byte, graphic, and mixed data. Specify this parameter only if your subsystem contains ASCII encoded data that is involved in bulk data movement. PowerExchange maps these ASCII code pages to equivalent EBCDIC code pages that are defined in the EBCDIC_CCSID parameter. Also include the EBCDIC_CCSID parameter.

Valid values are from 1 through 65534 for each subparameter. A value of 65534 indicates no CCSID.

The z/OS Installation Assistant inserts the values 850,65534,65534 unless you change them during installation.

UNICODE_CCSID=({*sbcsc_ccsid*|367},{*graphic_ccsid*|1200},{*mixed_ccsid*|1208})

Optional. The CCSIDs for Unicode single-byte, graphic, and mixed data. PowerExchange maps these Unicode code pages to equivalent EBCDIC code pages that are defined in the EBCDIC_CCSID parameter. Also include the EBCDIC_CCSID parameter.

Valid values are from 1 through 65534 for each subparameter. A value of 65534 indicates no CCSID.

The default values of 367,1208,1200 are used if you do not specify the DB2CODEPAGE statement or if you accepted the z/OS Installation Assistant values without modification. Usually, these default values are suitable.

Note: In DB2 Version 8.1 and later, the DB2 catalog tables are stored in UTF-8, which corresponds to CCSID 1208.

PLAN_CCSID=({*sbcsc_ccsid*|037},{*graphic_ccsid*|037},{*mixed_ccsid*|037})

Optional. The CCSIDs to use for EBCDIC single-byte, graphic, and mixed data instead of those in the EBCDIC_CCSID parameter. Use this parameter when you need to redirect the EBCDIC code pages to other EBCDIC code pages.

For example, use this parameter in the following situations:

- The EBCDIC_CCSID code pages do not have an ICU conversion table that PowerExchange can use for ICU-based code page conversion.

- The EBCDIC_CCSID code pages match the default code pages that were defined for the DB2 subsystem but differ from the EBCDIC code pages for a particular source or target table.

Default is 037,037,037.

REMAP n =(*current_data_ccsid*,*remapped_data_ccsid*)

Optional. If you specified DB2TRANS=R, you can use this parameter to have DB2 remap the code page in which the data is stored to another code page that you specify. For example, if you have ASCII data that does not map to the code page in which the DB2 plan is bound and that does not have an ICU convertor, use this parameter to remap the ASCII code page to a supported EBCDIC code page.

Alternatively, if you specified DB2TRANS=N, DB2 does not translate or remap the data. However, PowerExchange can use the REMAP n statement to substitute the correct code page for the incorrect one. For example, DB2 might report a data code page that does not match the code page defined in the DB2 catalog, possibly because the data was loaded incorrectly. In this case, you can specify the correct code page in the REMAP n parameter.

You can specify up to six REMAP n parameters in a DB2CODEPAGE statement, each for a different DB2 table. Increment the n number at the end of the parameter names so that each name is unique.

Usage Notes:

- If a PowerExchange Listener connects to a DB2 subsystem for which there is no DB2CODEPAGE statement, PowerExchange uses the code page of the PowerExchange Listener.
- During PowerExchange installation, a bind job binds the DB2 plan for PowerExchange as EBCDIC, without specifying the CCSIDs. As a result, PowerExchange uses the default application programming CCSIDs that were defined for the DB2 subsystem when it was created. PowerExchange retrieves these CCSIDs from the DB2 catalog tables and uses them along with the DB2CODEPAGE parameters to determine the code page to use.
- The values that you define for the DB2CODEPAGE statement must match the values that were specified for the DB2 subsystem on the application programming defaults panel, DSNTIPF. If you did not specify a value for the graphic and mixed CCSIDs in DB2, specify 65534 for the *graphic_ccsid* and *mixed_ccsid* parameters in the DB2CODEPAGE statement.

If the CCSIDs that you specified for the DB2 subsystem differ from the DB2CODEPAGE default CCSIDs, you must edit the DB2CODEPAGE statement to match the DB2 CCSIDs.

The following table shows the DB2 configuration options on the DSNTIPF panel and the DSNHDECP load module that correspond to the DB2CODEPAGE parameters:

DB2CODEPAGE Parameter	DSNTIPF Field	DSNHDECP Parameter
EBCDIC_CCSID	EBCDIC CCSID	SCCSID (single-byte), MCCSID (mixed), GCCSID (graphic)
ASCII_CCSID	ASCII CCSID	ASCCSID (single-byte), AMCCSID (mixed) AGCCSID (graphic)
UNICODE_CCSID	UNICODE CCSID	USCCSID (single-byte), UMCCSID (mixed), UGCCSID (graphic)
MIXED	MIXED DATA	MIXED

For more information about CCSID values and their meanings, see the *IBM DB2 for z/OS Installation Guide* for the DB2 version.

- If you click **Advanced Parms** on the **DB2 Parameters** page of the z/OS Installation Assistant, you can enter CCSID values for the following parameters:

- DB2CODEPAGE_ASCII_DBCS_CCSID
- DB2CODEPAGE_ASCII_MIXED_CCSID
- DB2CODEPAGE_ASCII_SBCS_CCSID
- DB2CODEPAGE_EBCDIC_DBCS_CCSID
- DB2CODEPAGE_EBCDIC_MIXED_CCSID
- DB2CODEPAGE_EBCDIC_SBCS_CCSID
- DB2CODEPAGE_MIXED
- DB2CODEPAGE_UNICODE_DBCS_CCSID
- DB2CODEPAGE_UNICODE_MIXED_CCSID
- DB2CODEPAGE_UNICODE_SBCS_CCSID

These installation parameters populate the EBCDIC_CCSID, ASCII_CCSID, and UNICODE_CCSID parameters in the DB2CODEPAGE statement of the DBMOVER file. You can edit them in the DBMOVER file if necessary.

- PowerExchange automatically generates a minimal DB2CODEPAGE specification that includes the UNICODE_CCSID and EBCDIC_CCSID parameters if no DB2CODEPAGE parameters are defined. The UNICODE_CCSID is included because DB2 catalog tables use Unicode encoding.
- The DB2CODEPAGE statement applies to bulk data movement operations only. For DB2 CDC, PowerExchange always uses the native code page of the DB2 data.
- The DB2CODEPAGE statement does not affect the CODEPAGE statement. If you use the DB2DEF_ENCODING statement, in certain cases, you might need to edit the DB2CODEPAGE statement to set the *mixed_ccsid* value to 037 in the ASCII_CCSID, EBCDIC_CCSID, or UNICODE_CCSID parameter that corresponds to the DB2DEF_ENCODING option.
- If PowerExchange tries to read columns that contain VARGRAPHIC or LOB data in a table on a single-byte subsystem, and the default DB2CODEPAGE setting of DB2TRANS=P is in effect, the you might receive the following SQL error:

```
-332 CHARACTER CONVERSION BETWEEN CCSID from_ccsid TO to_ccsid REQUESTED BY
      reason_code IS NOT SUPPORTED
```

If this situation occurs, specify DB2TRANS=N in the DB2CODEPAGE statement.

RELATED TOPICS:

- [“DB2 for z/OS Code Page Parameters” on page 350](#)

DB2DEF_ENCODING Statement

The DB2DEF_ENCODING statement defines the default encoding scheme that PowerExchange assigns to any DB2 columns that do not have an encoding scheme when you create a DB2 data map.

Operating Systems: z/OS

Data Sources: DB2 for z/OS

Related Statements: DB2CODEPAGE

Required: No

Syntax:

```
DB2DEF_ENCODING={A|E|U}
```

Valid Values:

- **A.** ASCII encoding.
- **E.** EBCDIC encoding.
- **U.** Unicode encoding.

Default is E.

Usage Notes:

- An encoding scheme might not be specified in the SYSIBM.SYSDATABASE table as a result of multiple DB2 migrations.
- In general, specify the encoding that is defined for DB2 in the **DEF ENCODING SCHEME** field on the application programming defaults panel, **DSNTIPF**. However, if DSNTIPF specifies 65534 for the mixed code page, edit the DB2CODEPAGE statement to set the *mixed_ccsid* value of the comparable ASCII, EBCDIC, or Unicode parameter to 037. The 037 value is required for PowerExchange to read the DB2 catalog tables.

For example, if you use the default value of E for the DB2DEF_ENCODING statement and the DSNTIPF panel specifies 65534 for the EBCDIC mixed code page, specify 037 as the last positional subparameter of the EBCDIC_CCsid parameter in the DB2CODEPAGE statement:

```
EBCDIC_CCsid=(sbcsc_ccsid,graphic_ccsid,037)
```

DB2ID Statement

The DB2ID statement defines the DB2 subsystem, plan, and access method load module that PowerExchange uses to process data from a DB2 for z/OS source for bulk data movement.

Operating Systems: z/OS

Data Sources: DB2 for z/OS

Related Statements: DB2PLAN

Required: No

Syntax:

```
DB2ID=(db2_subsystem
      [,plan]
      [{module_name|DTLAMV8F}]
)
```

Parameters:***db2_subsystem***

The DB2 subsystem identifier. The value is populated at installation. If you are creating a DB2ID statement for another DB2 subsystem, you must enter this value.

plan

The DB2 plan name for PowerExchange bulk data movement operations. Default is the value from the DB2PLAN statement.

{module_name|DTLAMV8F}

The PowerExchange DB2 access method load module. Options are:

- **DEFAULT.** The default access method load module, which is DTLAMV8F.
- **DTLAMV8F.** The module that PowerExchange uses to process multiple rows of data at a time by using DB2 multiple-row FETCH and INSERT SQL statements.

Default is DTLAMV8F.

Usage Notes:

- If you can use the defaults for all parameters in this statement, including the subsystem ID from installation, you do not need to specify the DB2ID statement. Otherwise, you must define a DB2ID statement.
- You can specify up to 25 DB2ID statements in a DBMOVER member.

DB2PLAN Statement

The DB2PLAN statement defines the DB2 plan that PowerExchange uses for bulk data movement processing.

Operating Systems: z/OS

Data Sources: DB2 for z/OS

Related Statements: DB2ID

Required: No

Syntax:

```
DB2PLAN={plan|DTLPLvrm}
```

Value: For the *plan* variable, enter the DB2 plan name for PowerExchange bulk data movement operations. Default is DTLPLvr*m*, where *vr*m** is the numeric value for the PowerExchange version, release, and modification level. For example, for PowerExchange 8.6.1, the *vr*m** value is 861. For PowerExchange 9.5.0, the *vr*m** is 950.

Usage Notes:

- PowerExchange uses the plan name from the DB2PLAN statement to access DB2 subsystems that you do not define with a DB2ID statement, or for which you do not specify a plan name in a DB2ID statement. If you use unique plan names for PowerExchange in different DB2 subsystems, define the subsystems and their unique plan names in the DB2ID statement.
- During the installation process, PowerExchange customizes the DBMOVER member and the XIDDB210 bind job with the plan name you specify in the z/OS Installation Assistant.

DECPOINT Statement

The DECPOINT statement defines the character that PowerExchange uses as the decimal point character in fields that contain noninteger numbers.

Data Sources: All

Required: No

Syntax:

```
DECPOINT={"character"|_}
```

Value: For the *character* variable, enter a single character enclosed in double quotation marks. Default is the period (.) character.

Example: If you use the period (.) character, which is the default character, PowerExchange writes noninteger numbers in the following format:

```
129.95
```


DEFAULTCHAR Statement

The DEFAULTCHAR statement defines the character that PowerExchange uses to replace any character data in a nonrelational data source that is not valid.

Data Sources: Nonrelational

Required: No

Syntax:

```
DEFAULTCHAR={character"|*}
```

Value: For the *character* variable, enter a single character enclosed in double quotation marks. Default is the asterisk (*) character.

DEFAULTDATE Statement

The DEFAULTDATE statement defines a numeric date string that PowerExchange uses to replace any missing component in a date field.

Data Sources: All

Required: No

Syntax:

```
DEFAULTDATE={yyyymmdd|19800101}
```

Value: For the *yyyymmdd* variable, enter a one- to eight-character numeric string. Default is 19800101.

DISABLE_PARTITIONS Statement

The DISABLE_PARTITIONS statement controls whether PowerExchange disables partitioning of the source data for bulk data movement operations.

Operating Systems: All

Data Sources: Offloaded DB2 unload, sequential, and VSAM data sets

Required: No

Syntax:

```
DISABLE_PARTITIONS={N|Y}
```

Valid Values:

- **N.** Enables partitioning of the source data. For offloaded DB2 unload, VSAM, and sequential data sources, if you specify the same or no SQL override for each partition, PowerExchange reads the source data one time and distributes data across the partitions.

Unless you have a specific reason to disable partitioning and read the source data multiple times, use the default of N for the DISABLE_PARTITIONS statement.

- **Y.** Disables partitioning of the source data for offloaded DB2 unload, sequential, and VSAM data sets, which is the default behavior for PowerExchange releases earlier than 9.0. If you specify the same or no SQL override for each partition, PowerExchange reads the source data multiple times instead of one time. If you set the **Worker Threads** connection attribute to a value greater than 0, PowerExchange uses multiple worker threads for the session.

Default is N.

Usage Notes: Use the DISABLE_PARTITIONS statement to preserve existing behavior when upgrading from a PowerExchange release earlier than 9.0 to 9.0 or later. The statement provides backward compatibility in special cases. For more information, see the *PowerExchange Installation and Upgrade Guide*.

DISP Statement

The DISP statement specifies the status, normal disposition, and abnormal disposition that PowerExchange uses when dynamically allocating target data sets on z/OS, if you do not provide DISP information.

The parameters and options on the DISP statement are equivalent to the options on the DISP parameter of the JCL DD statement.

Operating Systems: z/OS

Related Statements: BS, LRECL, RECFM, RELEASE, SPACE, UNIT, and VOLSER

Required: No

Syntax:

```
DISP=(status
      [,normal_disp]
      [,abnormal_disp]
)
```

Parameters:

status

Required. Data set status. Enter one of the following options:

- **M.** MOD.
- **N.** NEW.
- **O.** OLD.
- **S.** SHR.

Default is S.

normal_disp

Optional. Normal termination disposition for the data set. Enter one of the following options:

- **C.** CATLG.
- **D.** DELETE.
- **K.** KEEP.
- **P.** PASS.

Default is K.

abnormal_disp

Optional. Abnormal termination disposition for the data set. Enter one of the following options:

- **C.** CATLG.
- **D.** DELETE.
- **K.** KEEP.
- **P.** PASS.
- **U.** UNCATLG.

Default is K.

DM_RESOURCE Statement

The DM_RESOURCE statement specifies the suffix that PowerExchange uses to create the PowerExchange-specific resource profile that controls access to the DATAMAPS data set.

Operating Systems: z/OS

Related Statements: DM_SUBTASK, RACF_CLASS, and SECURITY

Required: No

Syntax:

```
DM_RESOURCE={resource_name|DATASET}
```

Value: For the *resource_name* variable, enter the suffix that PowerExchange appends to DTL.DATAMAP. to create the PowerExchange-specific resource profile. Default is DATASET.

Usage Notes:

- By default, PowerExchange checks the DTL.DATAMAP.DATASET resource profiles in the FACILITY class. The profile name and class might be different if you specify other values on the DM_RESOURCE and RACF_CLASS statements.
- To authorize access to the DATAMAPS data set, complete the following tasks:
 1. Specify Y for the DM_SUBTASK statement.
 2. Specify 2 for the first parameter of the SECURITY statement.
 3. Create the resource profile in your security system.
 4. On z/OS, grant access to the resource profile to all users that read or write data maps.

RELATED TOPICS:

- [“Security for Data Maps ” on page 273](#)

DM_SUBTASK Statement

The DM_SUBTASK statement controls how the PowerExchange Listener accesses the file that stores data maps.

Use this statement to reduce overhead in an environment where multiple PowerExchange Listeners are running and make frequent open and close requests on the data map file.

Operating Systems: z/OS

Related Statements: DATAMAP_SERVER, DM_RESOURCE, DMXCACHE_DELETEECSA, DMXCACHE_MAX_MEMORY_MB, DMXCACHE_MULTIPLEJOBS, RACF_CLASS, and SECURITY

Required: No

Syntax:

```
DM_SUBTASK={N|R|Y}
```

Valid Values:

- **N.** PowerExchange opens and closes the DATAMAPS data set in PowerExchange Listener subtasks, as required.
- **R.** To improve the read performance for data maps, the PowerExchange Listener starts a subtask that opens the DATAMAPS data set in read mode. No change in processing or performance occurs for insertion and deletion of data maps.

The R option is faster than the N option but slower than the Y option.

- **Y.** A single PowerExchange subtask owns and opens the DATAMAPS data set. All other subtasks use the data map subtask to access data maps. The DATAMAPS data set remains open until the PowerExchange Listener is shut down.

Enter Y to decrease resource utilization and improve the performance of the PowerExchange Listener. PowerExchange opens that DATAMAPS data set one time during the life of a PowerExchange Listener address space instead of opening and closing the data set for each request.

If you enter Y and also define the DATAMAP_SERVER statement, PowerExchange designates the PowerExchange Listener as a data map server.

Default is N.

Usage Notes:

- If you specify DM_SUBTASK=Y and also specify 2 for the first parameter of the SECURITY statement, PowerExchange checks this resource profile to determine whether to permit access to the DATAMAPS data set. In this case, you must code the resource profile in your security system and grant access to all users that read or write data maps on z/OS.

By default, PowerExchange checks the DTL.DATAMAP.DATASET resource profiles in the FACILITY class. The profile name and class might be different if you specify other values on the DM_RESOURCE and RACF_CLASS statements.

- If you specify DM_SUBTASK=Y and a PowerExchange Listener holds an exclusive lock on the DATAMAPS data set, also enter DATAMAP_SERVER=*node_name* to enable other tasks, such as the IMS synchronous ECCR or a netport job, to access the data set. Otherwise, the tasks fail.

Alternatively, use data maps caching. Informatica recommends this approach to improve performance and to avoid access problems that can occur if the PowerExchange Listener data maps server stops. To implement data maps caching, specify DM_SUBTASK=R and DMXCACHE_MAX_MEMORY_MB=20. With these settings, the PowerExchange Listener opens the data set in read-only mode and allows other tasks to access the data set.

RELATED TOPICS:

- [“Security for Data Maps ” on page 273](#)

DMXCACHE_DELETEECSA Statement

The DMXCACHE_DELETEECSA statement determines whether PowerExchange frees ECSA memory when data maps caching runs in multiple-jobs mode and no files exist with nonzero use counts in ECSA memory.

Operating Systems: z/OS

Related Statements: DM_SUBTASK, DMXCACHE_MAX_MEMORY_MB, and DMXCACHE_MULTIPLEJOBS

Required: No

Syntax:

```
DMXCACHE_DELETEECSA={N|Y}
```

Valid Values:

- **N.** PowerExchange does not free ECSA memory. Because allocating shared memory is a complex task and 4096 bytes of memory is a small amount, you can accept the default value, which is N, to permanently retain the ECSA memory.

- **Y.** PowerExchange frees ECSA memory when data maps caching runs in multiple-jobs mode and no files exist with nonzero use counts in ECSA memory.

To run data maps caching in multiple-jobs mode, set the DMXCACHE_MULTIPLEJOBS statement to Y.
Default is N.

DMXCACHE_MAX_MEMORY_MB Statement

The DMXCACHE_MAX_MEMORY_MB statement enables PowerExchange data maps caching and configures the maximum size for the data maps cache in megabytes.

Operating Systems: z/OS

Related Statements: DM_SUBTASK, DMXCACHE_DELETEECSA, and DMXCACHE_MULTIPLEJOBS

Required: No

Syntax:

```
DMXCACHE_MAX_MEMORY_MB={cache_size|0}
```

Valid Values: For the *cache_size* variable, enter the maximum size for the data maps cache in megabytes. Valid values are from 0 through 4095. You must enter a value greater than 0 to enable data maps caching. If you accept the default value of 0, PowerExchange does not use data maps caching.

Default is 0.

Notes:

- Verify that the REGION size specified in the DTLLST step in the JCL is large enough to run the PowerExchange Listener job. For example, if you define DMXCACHE_MAX_MEMORY_MB=20, you might have to increase the REGION size by 20 MBs.
- You cannot use data maps caching with IMS netport jobs.

DMXCACHE_MULTIPLEJOBS Statement

The DMXCACHE_MULTIPLEJOBS statement determines whether PowerExchange runs data maps caching in multiple-jobs or single-job mode.

Operating Systems: z/OS

Related Statements: DM_SUBTASK, DMXCACHE_DELETEECSA, and DMXCACHE_MAX_MEMORY_MB

Required: No

Syntax:

```
DMXCACHE_MULTIPLEJOBS={N|Y}
```

Valid Values:

- **N.** PowerExchange runs data maps caching in single-job mode.
- **Y.** PowerExchange runs data maps caching in multiple-jobs mode.

Default is N.

Single-Job Mode: PowerExchange maintains copies of previously read data maps in job-level memory. On subsequent reads of data maps, PowerExchange retrieves the data maps from job-level memory, which eliminates the overhead of enqueues, opens, points, reads, and closes of the data maps KSDS file.

PowerExchange does not use ESCA memory to maintain information about data maps.

When a data map changes or is deleted, PowerExchange deletes the copy of that data map in the cache. For changed data maps, PowerExchange does not add the data map to the cache again until the data map is actually used.

Single-job mode is more efficient than multiple-jobs mode because in single-job mode, the PowerExchange Listener job is the only job that updates the data maps file and the data maps cache does not become stale. Consequently, PowerExchange never needs to completely clear the cache like it does in multiple-jobs mode.

Multiple-Jobs Mode: In multiple-jobs mode, PowerExchange maintains copies of previously read data maps in job-level memory. Additionally, when you start a PowerExchange Listener, PowerExchange dynamically allocates 4096 bytes of ECSA memory in which PowerExchange maintains the name of the data maps KSDS file and the time stamp of its last update. PowerExchange uses this information to determine whether the data maps cache is stale and needs to be cleared.

On subsequent reads of data maps, PowerExchange retrieves the data maps from job-level memory, which eliminates the overhead of enqueues, opens, points, reads, and closes of the data maps KSDS file.

When a data map changes or is deleted, PowerExchange completes the following processing:

- In ECSA memory, PowerExchange changes the time stamp of the last update to the data maps KSDS file.
- When another task tries to access any data map, PowerExchange determines if the cache is stale by comparing the time stamp in ECSA memory against the time stamp of the data maps cache. If the cache is stale, PowerExchange clears it, and then PowerExchange reads the required data map from disk and adds it to the cache.

When you close a PowerExchange Listener, PowerExchange frees the ECSA memory if the following conditions are true:

- The DMXCACHE_DELETEECSA statement is set to Y in the DBMOVER configuration file.
- No files exist with nonzero use counts in the ECSA memory.

When you run data maps caching in multiple-jobs mode, you can optionally use the PWXUDMX utility to manage the ECSA memory. For more information about the utility, see the *PowerExchange Utilities Guide*.

DMX_DIR Statement

The DMX_DIR statement specifies the location that PowerExchange uses to read and store data maps.

Operating Systems: All

Required: No

Syntax:

```
DMX_DIR=location
```

Value: For the *location* variable, enter one of the following values, based on the operating system:

- **i5/OS.** The library name for the data maps. Maximum length is 10 characters.
Default is STDATAMAPS.
- **Linux, UNIX, and Windows.** The fully qualified path that contains the data maps. Maximum length is 512 characters.
On Linux and UNIX, default is ./datamaps.
On Windows, default is .\datamaps.
- **z/OS.** The DD statement name from the JCL that points to the DATAMAPS data set. Maximum length is eight characters.
Default is DATAMAP.

DTLMSG_CODEPAGE Statement

The DTLMSG_CODEPAGE statement specifies the code page identifier that indicates which language version of the PowerExchange messages file to use.

If you specify DTLMSG_CODEPAGE=UTF8, you must also specify the DTLMSG_LANGUAGE statement to indicate the language.

To determine the name of the messages file, PowerExchange concatenates the following elements:

```
dtlmsg + [_code_page + [_language]] + .txt
```

Data Sources: All

Related Statements: DTLMSG_LANGUAGE, LOG_CODEPAGE, CODEPAGE

Required: No

Syntax:

```
DTLMSG_CODEPAGE=code_page
```

Value: For the *code_page* variable, enter a code page identifier. Default is blank, which selects the UTF-8 English-language messages file, dtlmsg.txt.

Examples:

- To use the UTF-8 English-language version of the messages file, dtlmsg.txt, accept the default value by not defining the DTLMSG_CODEPAGE statement.
- To use the localized Japanese messages file, define the following statement:

```
DTLMSG_CODEPAGE=SHIFT-JIS
```

PowerExchange uses the multibyte version of the messages file, dtlmsg_SHIFT-JIS.txt.

- To use the localized Brazilian Portuguese messages file, define the following statements:

```
DTLMSG_CODEPAGE=UTF8  
DTLMSG_LANGUAGE=PORTUGUESE-BZ
```

PowerExchange uses the UTF-8 Brazilian Portuguese version of the messages file, dtlmsg_UTF8_PORTUGUESE-BZ.txt.

DTLMSG_LANGUAGE Statement

The DTLMSG_LANGUAGE statement specifies the language that, along with a DTLMSG_CODEPAGE identifier of UTF8, indicates which version of the PowerExchange messages file to use.

Operating Systems: Linux, UNIX, and Windows

Related Statements: DTLMSG_CODEPAGE

Required: No

Syntax:

```
DTLMSG_LANGUAGE=language
```

Example: To use the UTF-8 Brazilian Portuguese messages file, dtlmsg_utf8_portuguese-bz.txt, define the following statements:

```
DTLMSG_CODEPAGE=UTF8  
DTLMSG_LANGUAGE=PORTUGUESE-BZ
```

ENABLE_AUTOCOMMIT Statement

The ENABLE_AUTOCOMMIT statement controls whether PowerExchange commits data to the target each time the data buffer becomes full during a bulk data movement session.

Operating Systems: All

Required: No

Syntax:

```
ENABLE_AUTOCOMMIT={N|Y}
```

Valid Values:

- **N.** PowerExchange commits data to the target only at each commit interval during a bulk data movement session.
- **Y.** PowerExchange commits data to the target each time the data buffer becomes full or the commit interval that is defined in the session properties is reached, whichever occurs first. You set the data buffer size for a session by defining the **Default Buffer Block Size** session property in the **Advanced** settings of the **Config Object** tab.

Default is N.

Caution: Avoid using this statement if possible. Defining ENABLE_AUTOCOMMIT=Y alters the standard commit and rollback behavior for PowerCenter sessions. Instead of this statement, use one of the following methods to avoid out-of-memory errors during bulk data movement sessions:

- Reduce the **Commit Interval** sessions property.
- Based on the error message that you receive, tune the database.

ENCRYPT Statement

The ENCRYPT statement controls whether PowerExchange uses encryption when moving data. In addition to an internal proprietary encryption method, PowerExchange supports AES encryption. The ENCRYPT statement provides an alternative to Secure Sockets Layer (SSL) authentication.

Note: Informatica recommends that you use SSL authentication instead of the ENCRYPT statement. SSL authentication provides stricter security and is used by other Informatica products.

Operating Systems: All

Related Statements: ENCRYPTLEVEL and SSL

Required: No

Syntax:

```
ENCRYPT={AES|N|Y}
```

Valid Values:

- **AES.** PowerExchange encrypts data by using the Advanced Encryption Standard (AES) algorithm. You must specify the encryption level by coding the ENCRYPTLEVEL statement.
- **N.** PowerExchange does not encrypt data.
- **Y.** PowerExchange uses its internal proprietary encryption method to encrypt data.

Default is N.

Usage Notes:

- To avoid confusion, specify the same value in the ENCRYPT statement in the DBMOVER configuration files on the client and PowerExchange Listener machines. Also, specify the same value in the ENCRYPTLEVEL statement in the DBMOVER configuration file on both machines. However, PowerExchange always uses the ENCRYPT and ENCRYPTLEVEL values in the DBMOVER configuration file on the client machine that initiates the connection to the PowerExchange Listener.
- The ENCRYPT and ENCRYPTLEVEL statements control encryption of data that PowerExchange moves between the PowerExchange Listener and a client such as the PowerExchange Navigator or a PowerExchange utility. These statements also control encryption of data that PowerExchange moves between the PowerExchange Listener and the Integration Service for connections that have the default **Encryption Type** setting of **None**. For connections with an **Encryption Type** setting of **AES**, the **Encryption Type** and **Encryption Level** connection attributes override the settings in the DBMOVER configuration file.

RELATED TOPICS:

- [“Secure Sockets Layer Support” on page 300](#)

ENCRYPTLEVEL Statement

The ENCRYPTLEVEL statement specifies the encryption level to use if you select AES encryption. The encryption level determines the encryption key length.

If you enter AES in the ENCRYPT statement, you must include the ENCRYPTLEVEL statement.

Operating Systems: All

Related Statements: ENCRYPT

Required: Yes if you use AES encryption

Syntax:

```
ENCRYPTLEVEL={1|2|3}
```

Valid Values:

- **1.** Use a 128-bit encryption key.
- **2.** Use a 192-bit encryption key.
- **3.** Use a 256-bit encryption key.

Default is 1.

Usage Notes:

- To avoid confusion, specify the same value in the ENCRYPT statement in the DBMOVER configuration files on the client and PowerExchange Listener machines. Also, specify the same value in the ENCRYPTLEVEL statement in the DBMOVER configuration file on both machines. However, PowerExchange always uses the ENCRYPT and ENCRYPTLEVEL values in the DBMOVER configuration file on the client machine that initiates the connection to the PowerExchange Listener.
- The ENCRYPT and ENCRYPTLEVEL statements control encryption of data that PowerExchange moves between the PowerExchange Listener and a client such as the PowerExchange Navigator or a PowerExchange utility. These statements also control encryption of data that PowerExchange moves between the PowerExchange Listener and the Integration Service for connections that have the default **Encryption Type** setting of **None**. For connections with an **Encryption Type** setting of **AES**, the **Encryption Type** and **Encryption Level** connection attributes override the settings in the DBMOVER configuration file.

ENQMAJORNAME Statement

The ENQMAJORNAME statement specifies the major name for ENQ macros that PowerExchange issues to lock PowerExchange data sets and resources.

Operating Systems: z/OS

Related Statements: ENQSYSTEMS

Required: No

Syntax:

```
ENQMAJORNAME={major_name|DETAIL}
```

Value: For the *major_name* variable, enter the major name for ENQ macros that PowerExchange issues to lock PowerExchange data sets and resources, such as the PowerExchange message logs, capture registrations, and DB2 bulk load operations. Default is DETAIL.

Usage Notes:

- By default, the scope of ENQ macros that PowerExchange issues is SYSTEM, which means PowerExchange locks data sets within a single system. To change the scope of the ENQ macros that PowerExchange issues to SYSTEMS, specify Y for the ENQSYSTEMS statement.
- For data set locks, PowerExchange uses the data set name as the minor name in the ENQ macros.
- For resource locks, PowerExchange uses minor names related to the resource.

The following table lists the minor names that PowerExchange uses to protect resources:

ENQ Minor Name	Usage Description
DTLAMCPR	Capture registrations operations.
DTLCACDC	CDCT operations.
DTLDB2	DB2 LOAD utility operations.
DTLILCAT	Updates to the log catalog for CA IDMS/DB log-based CDC.
DTLLOG	Updates to the standard PowerExchange message log, DTLLOG.

ENQSYSTEMS Statement

The ENQSYSTEMS statement controls whether PowerExchange uses a scope of SYSTEM or SYSTEMS for ENQ macros that it issues to lock PowerExchange data sets and resources.

Operating Systems: z/OS

Related Statements: ENQMAJORNAME

Required: No

Syntax:

```
ENQSYSTEMS={N|Y}
```

Valid Values:

- **N.** PowerExchange uses a scope of SYSTEM on ENQ macros, which provides protection on the system on which the ENQ is issued.

- **Y.** PowerExchange uses a scope of SYSTEMS on ENQ macros. To provide multisystems protection with these enqueues, you must use GRS or a similar product to propagate them to other systems.

Warning: If you specify Y for the ENQSYSTEMS statement and you propagate enqueues to other z/OS systems that all use the same major name, you might experience contention for some PowerExchange resources.

Default is N.

Usage Notes: By default, PowerExchange uses a major name of DETAIL. To change the major name, use the ENQMAJORNAME statement.

ERRROWNOTFOUND Statement

The ERRROWNOTFOUND statement specifies whether PowerExchange generates or does not generate errors for UPDATE or DELETE operations on nonexistent rows.

Include the ERRROWNOTFOUND statement in the DBMOVER configuration file on the target system.

Data Sources: All

Required: No

Syntax:

```
ERRROWNOTFOUND={N|Y}
```

Valid Values:

- **N.** PowerExchange does not generate errors.
- **Y.** PowerExchange generates an error, increments the error count, and writes the record in error to the reject file.

Default is N.

Usage Notes: This statement is valid only with Asynchronous with Fault Tolerance write mode. To use this mode, set the **Write Mode** value to **Asynchronous with Fault Tolerance** in the PWXPC Connection attribute.

EXT_CP_SUPPT Statement

The EXT_CP_SUPPT statement controls character conversion in single-byte static code pages.

This statement controls whether PowerExchange converts certain characters from their EBCDIC to their corresponding ASCII values. The statement affects EBCDIC characters X'41', X'FF', and characters with a value of less than X'40'. X'40' represents the space character in an EBCDIC code page.

Operating Systems: All

Required: No

Syntax:

```
EXT_CP_SUPPT={N|Y}
```

Valid Values:

- **N.** PowerExchange does not convert the affected characters. For example, PowerExchange leaves the X'25' character in an EBCDIC source code page as X'25' in a target ASCII code page. In the ASCII code page, its meaning changes from a LINE FEED to a PERCENT SIGN (%).

- **Y.** PowerExchange converts any affected character to its Unicode meaning. For example, PowerExchange converts the X'25' character in the EBCDIC source code page IBM-037 to X'0A' in a target ISO-8859 code page. The Unicode name for this character is LINE FEED (LF).

Default is Y.

Usage Notes: The EXT_CP_SUPPT statement affects a small number of single-byte code pages. The statement applies only when both code pages involved have internal numbers ranging from 1 through 40. These code pages have the following names:

- HP-ROM8
- IBM-037
- IBM-273
- IBM-277
- IBM-278
- IBM-280
- IBM-284
- IBM-285
- IBM-297
- IBM-424
- IBM-500
- IBM-870
- IBM-1047
- ISO-8859
- MS-1250
- PC-856
- USRCP00

In PowerExchange releases earlier than 9.6.0, the EXT_CP_SUPPT default is N. In PowerExchange 9.6.0 and later, the default is Y. If you need to retain the previous default method of mapping for particular EBCDIC values (for example, if you need to map EBCDIC X'FF' to ASCII X'FF'), Informatica recommends that you create a customized ICU code page.

RELATED TOPICS:

- [“PowerExchange Globalization” on page 346](#)

FILEMAPPED_MEMORY_DIR Statement

The FILEMAPPED_MEMORY_DIR statement specifies the directory that stores temporary file-mapped memory files for PowerExchange Listener statistics on AIX PPC64, i5/OS (IBM i), and Solaris SP64 systems. If you set the STATS statement to MONITOR in the DBMOVER configuration file, the memory statistics are generated when either the optional monitoring interval in the MONITOR parameter elapses or when you issue the DISPLAYSTATS or pwxcmd displaystats command.

Informatica strongly recommends that you use this statement to specify a directory that is dedicated to the storage of temporary file-mapped memory files for PowerExchange Listener statistics. Specify a directory that is on a locally mounted disk for better performance and to avoid complications that result from network mountings, such as name clashes, disk usage problems, and the unnecessary inclusion of NFS files.

Operating Systems: AIX, i5/OS (IBM i), and Solaris

Related Statements: STATS

Required: No

Syntax:

```
FILEMAPPED_MEMORY_DIR=path/directory
```

By default, temporary file-mapped memory statistics files are written to the current directory of the user, or if the current directory is not available, to the top-most root directory of the file system.

Usage Notes:

- Monitor the directory that the FILEMAPPED_MEMORY_DIR statement specifies. On AIX or Solaris, you can use the `rm` command to delete the files that are not being used by an active PowerExchange process.
- If you cancel a Listener task by using an operating system command such as `ENDJOB *IMMED` or `kill -9`, the memory files are orphaned.

GDGLOCATE Statement

The GDGLOCATE statement controls whether the PowerExchange Listener uses the most recent z/OS catalog information to determine the latest generation in a generation data group (GDG) based on a relative generation number when reading or writing generations in a GDG.

Operating Systems: z/OS

Related Statements: NETPORT, TAPEWAIT, and WAITDSN

Required: No

Syntax:

```
GDGLOCATE={Y|N}
```

Valid Values:

- **Y.** The PowerExchange Listener uses the most recent z/OS catalog information for a generation data set when reading or writing a generation data set in a GDG based on a relative generation number.

PowerExchange refreshes the generation table for a GDG by specifying the S99GDGNT bit in S99FLAGS1 on a dynamic allocation (SVC99) request. The dynamic allocation request then uses the LOCATE function to determine the latest absolute generation in the GDG based on the catalog information.
- **N.** The PowerExchange Listener does not refresh the generation table with the latest information from the z/OS catalog after it references the GDG the first time by using a relative generation number. All future Listener requests that use the relative generation number will access the same absolute generation as the first reference.

Default is Y.

Usage Notes: Assume that you set GDGLOCATE to N, and the relative generation number of 0 for MY.GDG corresponds to the MY.GDG.G0011V00 data set. The first time you reference relative generation number 0 through the PowerExchange Listener, the Listener reads the absolute generation G0011V00. If you create a new generation that has an absolute generation number of G0012V00 and request relative generation 0 through the same PowerExchange Listener, the Listener still reads the absolute generation G0011V00. However, if you use the default setting of GDGLOCATE=Y and request relative generation 0, the Listener uses the latest z/OS catalog information to read the absolute generation G0012V00.

RELATED TOPICS:

- [“Using Netport Jobs for Generation Data Sets” on page 216](#)

GSBUFSIZE Statement

The GSBUFSIZE statement enables you to increase the maximum internal buffer size for PowerExchange processing of a row of data when the default buffer size is not adequate. Add this statement if you receive error message PWX-03034.

Operating Systems: All

Required: No

Syntax:

```
GSBUFSIZE={buffer_size|32768}
```

Value: For the *buffer_size* variable, enter the maximum internal buffer size, in bytes. This value should be at least twice the size of the longest row in your data sources. Valid values are from 16384 through 8388608. Default is 32768.

Usage Notes: Usually, the default value is sufficient. If you get the following error message, add this statement with a value greater than the default:

```
Group Source type record is too long: length_in_bytes
```

This message reports the length of the row that exceeded the buffer size. Ensure that the GSBUFSIZE value is at least twice this length.

HOSTNAME Statement

The HOSTNAME statement specifies an i5/OS host name that overrides the host name that PowerExchange retrieves from the TCP/IP Host table for communications with the PowerExchange Listener on i5/OS.

This statement pertains to an ethernet-type of environment in which an i5/OS server is configured to use multiple i5/OS host names and IP addresses. You might use this type of environment to improve performance when a high volume of transactions exists or to use different host names for different business functions. In this situation, the HOSTNAME statement is required.

Operating Systems: i5/OS

Related Statements: LISTENER

Required: No

Syntax:

```
HOSTNAME=host_name
```

Value: For the *host_name* variable, enter a valid TCP/IP host name up to 64 characters in length. No default value is provided.

Usage Notes: You can enter one HOSTNAME statement in the DBMOVER member in the CFG file on the i5/OS system. This statement can specify only one host name override.

Use the HOSTNAME statement when you want to start or connect to a PowerExchange Listener on i5/OS under a host name other than the one PowerExchange retrieves by means of a TCP/IP gethostname() call to the TCP/IP Host table and other than the one that the HOSTENT utility reports. PowerExchange tasks and utilities can then use the override host name to connect to the PowerExchange Listener on i5/OS to perform tasks such as fetching data or pinging the Listener. The specified host name is used in conjunction with the

IP address in the LISTENER statement, provided that the IP address is not associated with the host name returned by the TCP/IP gethostname() call.

ICUALIAS Statement

The ICUALIAS statement adds or deletes an alias that points to a custom ICU code page.

Operating Systems: All

Related Statements: ICUCNVPROPERTY, ICUCONVERTER, and ICUDATADIR

Required: No

Syntax:

```
ICUALIAS=(CPN
          ,command
          [,alias]
        )
```

Parameters:

CPN

Required. The code page slot number to which to add an alias, or from which to delete an alias. Valid values are from 301 through 340.

To identify available code page slots, see the ICUCHECK report.

command

Required. The command to apply to the specified code page slot number. Enter one of the following options:

- **ADD.** Adds the specified alias to point to the specified code page slot number.
For example, to add an alias of CP930 to point to CPN 302, include the following statement:

```
ICUALIAS=(302,ADD,CP930)
```

This command causes DB2 columns with CCSID 930 to use this custom ICU code page.

- **DELETE.** Deletes the specified alias from the specified code page slot number.
For example, to delete the CP930 alias from CPN 206, include the following statement:

```
ICUALIAS=(206,DELETE,CP930)
```

- **DELETEALL.** Deletes all aliases from the specified code page slot number.

The DELETEALL command uses the following parameters only:

```
ICUALIAS=(CPN,DELETEALL)
```

For example, to delete all defined aliases from CPN 206, include the following statement:

```
ICUALIAS=(206,DELETEALL)
```

alias

Required if the second parameter is ADD or DELETE. An alternative name for the code page, which can be used to find a code page. Aliases for DB2 CCSIDs begin with `cp`. Maximum length is 30 characters.

ICUCNVPROPERTY Statement

The ICUCNVPROPERTY statement sets customized properties for an ICU code page.

Note: Use the ICUCNVPROPERTY parameter in rare situations.

Operating Systems: All

Related Statements: ICUALIAS, ICUCONVERTER, ICUDATADIR, and USE_TYPE1_FALLBACKS

Required: No

Syntax:

```
ICUCNVPROPERTY=(CPN
                  ,property
                  ,value
                )
```

Parameters:

CPN

Required. The code page slot number for which to define a property. Valid values are from 301 through 340.

To identify available code page slots, see the ICUCHECK report.

property

Required. Enter one of the following options:

- **ASCII.** Indicates that the characters 0 to 9 and A through Z match the 7-bit ASCII values, such as in code page US_ASCII. For example:

```
ICUCNVPROPERTY=(301,ASCII,ON)
```

- **BUFFER_INCREMENT.** PowerExchange allocates extra bytes for the buffer when converting to the code page. Enter a numeric value for this property.

For example:

```
ICUCNVPROPERTY=(301,BUFFER_INCREMENT,8)
```

- **EBCDIC.** Indicates that the characters 0 to 9 and A through Z match the standard EBCDIC values, such as in code page IBM-37.

For example:

```
ICUCNVPROPERTY=(301,EBCDIC,ON)
```

- **FIX_MIN_LENGTH.** ICU processing drops leading binary zeros when converting to double-byte code pages. PowerExchange creates all characters at the correct minimum length and inserts leading binary zeros as needed.

The Show a code page uses this property.

For example:

```
ICUCNVPROPERTY=(301,FIX_MIN_LENGTH,ON)
```

- **IBM420_END_OF_WORD_SPACES.** Adds a space after certain IBM420 characters that end words.

For example:

```
ICUCNVPROPERTY=(301,IBM420_END_OF_WORD_SPACES,ON)
```

- **LOWERCASE_NONSTANDARD.** On z/OS, four Japanese code pages include lowercase a through z characters that do not match the standard EBCDIC value that code page IBM-37 uses.

For these code pages, the client program converts the SQL to uppercase before PowerExchange converts the SQL to the z/OS code page. CP930 uses this property.

For example:

```
ICUCNVPROPERTY=(301,LOWERCASE_NONSTANDARD,ON)
```

- **POWERCENTER_NAME.** Defines a PowerCenter name for the code page. The name must match an entry in the list on the define relational sources page in the Workflow Manager.

On Windows, to use an ICU code page with ODBC, define a PowerCenter name for the code page.

For example:

```
ICUCNVPROPERTY=(301,POWERCENTER_NAME,"IBM EBCDIC Japanese")
```

- **REVERSE_EBCDIC_DIGITS.** Reverses consecutive sequences of the following bytes: F0, F1, F2, F3, F4, F5, F6, F7, F8, and F9.

For example:

```
ICUCNVPROPERTY=(301,REVERSE_EBCDIC_DIGITS,ON)
```

- **SQL_PARSABLE.** Indicates that the invariant SQL characters must match the standard values for the ASCII or EBCDIC type. Use this property to determine whether the ODBC layer needs to translate SQL before searching for literals like SELECT, FROM, or WHERE.

For example:

```
ICUCNVPROPERTY=(301,SQL_PARSABLE,ON)
```

- **STATE_BYTES.** PowerExchange allocates two extra bytes for the buffer when converting to the code page.

For example:

```
ICUCNVPROPERTY=(301,STATE_BYTES,ON)
```

- **UNKNOWN_ENDIANNES.** Indicates that the code page has a different integer endianness, based on the system.

For example:

```
ICUCNVPROPERTY=(301,UNKNOWN_ENDIANNES,ON)
```

In some situations, the system can determine the correct code page. For example, switch from UTF-16 to UTF-16LE on Windows. Generally, however, avoid such code pages.

- **UNSHAPE_ARABIC.** Converts Unicode characters derived from IBM420 “shaped” characters to characters for Windows 1256. Converts Unicode characters from 0xFE70 through 0xFE74 to equivalent 0x62xx characters. Converts LAM ALEF ligature characters from 0xFE75 through 0xFE7C to the Unicode 0x0644 character and another character.

For example:

```
ICUCNVPROPERTY=(301,UNSHAPE_ARABIC,ON)
```

- **USE_TYPE1_FALLBACKS.** Enables type 1 fallback mappings for the specified code page.

For example:

```
ICUCNVPROPERTY=(242,USE_TYPE1_FALLBACKS,ON)
```

Default is OFF.

To enable type 1 fallback mappings for all code pages, use the USE_TYPE1_FALLBACKS DBMOVER statement.

Note: If you specify a property that is not valid, an error message displays the list of valid properties.

value

Required. The value for the property, as follows:

- For binary properties, either ON or OFF.
For new customer ICU code pages, default is OFF.
- For BUFFER_INCREMENT, a numeric value.
For new customer ICU code pages, default is 0.

ICUCONVERTER Statement

The ICUCONVERTER statement adds a code page converter to or deletes a code page converter from an ICU code page.

Operating Systems: All

Related Statements: ICUALIAS, ICUCNVPROPERTY, and ICUDATADIR

Required: No

Syntax: For the ADD command:

```
ICUCONVERTER= (CPN
                ,ADD
                ,code_page_name
                ,min_bytes
                ,max_bytes
                ,space_hex
                )
```

For the DELETE commands:

```
ICUCONVERTER= (CPN
                , {DELETE | DELETEALL})
```

Parameters:

CPN

Required. The code page slot number for which to add or delete a code page converter. Valid values are from 301 through 340.

To identify available code page slots, see the ICUCHECK report.

command

Required. The command to apply to the specified code page slot number. Enter one of the following options:

- **ADD.** Adds the specified code page converter to the specified code page slot number. For example, to add the user_showa_01 code page into CPN 301, include the following statement:

```
ICUCONVERTER=(301,ADD,user_showa_01,2,2,0000)
```

In this example, all characters are two bytes. The hexadecimal space character is \X00\X00.

The following example adds the user_cp930 code page into CPN 302:

```
ICUCONVERTER=(302,ADD,user_cp930,1,2,40)
```

All characters in this example use either one or two bytes. The hexadecimal space character is \X40.

- **DELETE.** Deletes the specified code page converter from the specified code page slot number. For example, to delete a code page and all its aliases from the system, include the following statement:

```
ICUCONVERTER=(301,DELETE)
```

If you include a DELETE statement, also include an ADD statement to add a replacement custom ICU code page.

- **DELETEALL.** Deletes all code page converters from the specified code page slot number.

The DELETEALL command uses the following parameters:

```
ICUALIAS=(CPN,DELETEALL)
```

code_page_name

Required for the ADD command. Not used by the DELETE commands. The name of the code page converter. This name is also a component of the CNV file name.

min_bytes

Required for the ADD command. Not used by the DELETE commands. The minimum number of bytes for each character.

max_bytes

Required for the ADD command. Not used by the DELETE commands. The maximum number of bytes for each character.

space_hex

Required for the ADD command. Not used by the DELETE commands. A hexadecimal character that represents the space character. PowerExchange verifies that it can convert this character in UTF-8 during initialization. PowerExchange appends this character to CHAR columns where the data is short.

ICUDATADIR Statement

The ICUDATADIR statement defines the directory location for the binary CNV files that makeconv produces.

Operating Systems: All

Related Statements: ICUALIAS, ICUCNVPROPERTY, and ICUCONVERTER

Required: Yes if you include an ICUCONVERTER ADD statement

Syntax:

```
ICUDATADIR=icudatadir
```

Value: For the *icudatadir* variable, enter the directory location for the binary CNV files that makeconv produces. On all systems except z/OS, the directory name is case sensitive. For example, on Windows, you might define the following statement:

```
ICUDATADIR=c:\ucm\ShowaUcm
```

Usage Notes:

- If you specify the ICUDATADIR statement, Power Exchange validates all custom ICU code pages by opening them and converting the space character into UTF-8. This validation fails if one of the following events occur:
 - The U_FILE_ACCESS_ERROR error occurs, which indicates that PowerExchange cannot load the binary CNV file.
 - PowerExchange cannot convert the space character into any character in UTF-8.
- If you connect to PowerExchange from a PowerCenter Integration Service, PowerExchange must use the same ICU data directory that PowerCenter uses, and you must copy the CNV files to this directory.

The PowerCenter Integration Service initializes before the PowerExchange connection and the ICU data directory is set to where the resources for localization are.

Consequently, PowerExchange must use the same ICU data directory that PowerCenter uses.

To define the ICU data directory, complete the following steps:

1. Define the ICUDATADIR statement to point to the same ICU data directory that PowerCenter uses. This directory is the `server\bin` directory that contains the ICU resource files with masks like `*.EN.res` or `*.JA.res`.

For example, you might define the ICUDATADIR statement as follows:

```
ICUDATADIR=C:\zeus\win32\install\server\bin
```

2. Copy the CNV file to the ICU data directory. Depending on the integer endianness, the CNV file name might begin with the `pmicudt32` string appended with the `l` or `b` character.

For example, the converter named `ibm-937_P110-1999` is in the `pmicudl32l_ibm-937_P110-1999.cnv` or `pmicudl32b_ibm-937_P110-1999.cnv` file.

IMSBSDS Statement

The IMSBDDS statement identifies the high-level qualifier of the IMS bootstrap data set, which is required to retrieve DBD information in DBGEN format from the IMS catalog. This statement also indicates the order of locations, such as the IMS catalog and IMS DBDLIB library, to search for DBD information.

This statement is required for PowerExchange to use the IMS catalog API to retrieve DBD information in DBGEN format from the IMS catalog when the IMS control region is not running or when you use an IMS version earlier than IMS 15. PowerExchange uses DBD information in DBGEN format when you create capture registrations in the PowerExchange Navigator or perform IMS unload processing. This statement is also required for PowerExchange to use the IMS Catalog Library Builder Utility, DFS3LU00, to get DBD information in source format from the IMS catalog when you create data maps from the PowerExchange Navigator.

Note: Use of the IMS catalog is optional. By default, PowerExchange gets DBD information from the DBDLIB library.

Enter this statement in the DBMOVER configuration file on the PowerExchange Listener system.

Operating Systems: z/OS

Data Sources: IMS

Required: Required to get DBD information from the IMS catalog

Syntax:

```
IMSBSDS=(ims_ssid
          ,bootstrap_hlq
          [,search_preference]
)
```

Parameters:

ims_ssid

Required. The IMS subsystem ID (SSID) for the subsystem that contains the source objects. This value must match the IMS SSID value that is specified in the first positional parameter in an IMSID statement.

bootstrap_hlq

Required. The high-level qualifier of the IMS bootstrap data set that is in the IMS directory associated with the IMS catalog for the IMS SSID.

search_preference

Optional. The order in which PowerExchange searches locations for DBD metadata for source objects. Valid values are:

- **D.** First search the DBDLIB library and then search the IMS catalog, if available.
- **C.** First search the IMS catalog and then search the DBDLIB library.
- **O.** Search the IMS catalog only.

If you do not specify an option, PowerExchange searches only the DBDLIB library by default and does not use the IMS catalog.

IMSID Statement

The IMSID statement defines characteristics of the IMS source subsystem for PowerExchange IMS bulk data movement operations and IMS log-based CDC.

Operating Systems: z/OS

Data Sources: IMS

Required: Yes, for IMS bulk data movement operations and IMS log-based CDC

Syntax:

```
IMSID=(ims_ssid
      [,dbdlib]
      [,RECON=(recon1
               [,recon2]
               [,recon3])])
)
```

Parameters:

ims_ssid

The IMS subsystem ID (SSID).

If you import source DBD information from the IMS catalog, this *ims_ssid* value is required and must match the *ims_ssid* value in an IMSBSDL statement.

For IMS log-based CDC, this *ims_ssid* value must match the **Recon Identifier** value specified in the registration group.

If you create data maps that use the **IMS ODBA** access method, this value is required for PowerExchange to connect to the IMS subsystem to access data.

dbdlib

For IMS log-based or synchronous CDC, the DBDLIB library that contains the database descriptions (DBDs) to be used for defining IMS data maps if you do not import the DBD information from the IMS catalog. The library name is an alphanumeric string up to eight characters in length. This parameter is not required if you import source metadata from the IMS catalog.

RECON=(recon1,[recon2],[recon3])

Optional. The RECON data sets for IMS log-based CDC. If you include the RECON parameter, you must specify at least one data set name. You can enter each data set name on a separate line in the DBMOVER member.

Usage Notes:

- You can specify up to 10 IMSID statements in a DBMOVER member.
- PowerExchange does not require the IMSID statement to create data maps if you import DBD information for source objects from the DBDLIB library. However, when you perform IMS bulk data movement operations, PowerExchange matches the *ims_ssid* in the IMSID statement against the **IMS SSID** value in the data map to determine the DBDLIB data set to use. Also, for IMS log-based CDC, PowerExchange matches the *ims_ssid* value in the IMSID statement against the *ims_ssid* value in the **Recon Identifier** field in the registration group.
- If you create data maps by importing DBD information for source objects from IMS catalog, the *ims_ssid* in the IMSID statement is required to find a matching IMSBSDL statement. If you create a registration group for IMS log-based CDC, the *ims_ssid* value in the **Recon Identifier** field matches against the *ims_ssid* value in an IMSBSDL statement.

JOBCLASS Statement

The JOBCLASS statement specifies the job class that PowerExchange uses for CA IDMS/DB metadata and DB2 LOAD utility batch jobs.

Operating Systems: z/OS

Data Sources: CA IDMS/DB and DB2 for z/OS

Required: No

Syntax:

```
JOBCLASS={class|A}
```

Value: For the *class* variable, enter the job class that PowerExchange uses for the CLASS parameter of the JOB card of CA IDMS/DB metadata and DB2 LOAD utility batch jobs. Valid characters are asterisk (*), A through Z, and 0 through 9. Default is A.

LDAP_BASE Statement

The LDAP_BASE statement specifies the location within an LDAP directory from which to conduct a PowerExchange user search. If you specify LDAP as the third parameter in the SECURITY statement, include at least one LDAP_BASE statement in the DBMOVER configuration file. You can specify up to five LDAP_BASE statements.

LDAP searches the search-base locations in the order in which LDAP_BASE statements occur in the DBMOVER configuration file. The search stops as soon as the PowerExchange user is found, or the search continues through all LDAP_BASE locations if no PowerExchange user is found.

Operating Systems: Linux, UNIX, and Windows

Related Statements: LDAP_FILTER, LDAP_LOGIN_ATTRIBUTE, LDAP_SCOPE

Required: No

Syntax:

```
LDAP_BASE=("search_base")
```

Value: For the *search_base* variable, enter the distinguished name (DN) of an LDAP tree node from which to conduct a PowerExchange user search. Enclose the *search_base* value in parentheses and double quotation marks.

Usage Notes:

- If you do not define an LDAP_BASE statement, PowerExchange uses the default search base, which is an empty string. This search base will likely cause all searches to fail.
- If you use multiple LDAP_BASE statements, for best search performance, order the statements so that a PowerExchange user is likely to be found as soon as possible.

RELATED TOPICS:

- [“Controlling the LDAP Search” on page 283](#)

LDAP_BIND_DN Statement

The LDAP_BIND_DN statement specifies the identity of the LDAP search user.

Operating Systems: Linux, UNIX, and Windows

Related Statements: LDAP_BIND_EPWD, LDAP_BIND_PWD, LDAP_BIND_TIMEOUT, LDAP_SASL_MECH

Required: No

Syntax:

```
LDAP_BIND_DN="search_user_distinguished_name"
```

Value: For the *search_user_distinguished_name* variable, enter the distinguished name of an LDAP user with authority to connect to the LDAP server and conduct a search for the PowerExchange user. Enclose the *search_user_distinguished_name* value in double quotation marks (").

Usage Notes:

- If you do not define the LDAP_BIND_DN statement or the LDAP_SASL_MECH statement, the LDAP client does not attempt to perform a bind and instead performs unauthenticated operations.
- If you specify an empty string in the LDAP_BIND_DN statement, the LDAP client requests an anonymous bind. An anonymous bind is acceptable if the LDAP server has been configured to allow anonymous access.
- If you configure LDAP TLS to require client certification and specify a SASL mechanism of EXTERNAL, the LDAP_BIND_DN statement is ignored and the identity of the search user is determined from the client certificate and the certificate mapping of the LDAP server. For more information, see ["LDAP Transport Layer Security" on page 284](#).

RELATED TOPICS:

- ["LDAP Search User" on page 283](#)

LDAP_BIND_EPWD Statement

The LDAP_BIND_EPWD statement specifies an encrypted password for the LDAP search user.

Operating Systems: Linux, UNIX, and Windows

Related Statements: LDAP_BIND_DN, LDAP_BIND_PWD, LDAP_SASL_MECH

Required: No

Syntax:

```
LDAP_BIND_epwd=search_user_encrypted_password
```

Value: For the *search_user_encrypted_password* variable, enter an encrypted password for the LDAP search user that is identified in the LDAP_BIND_DN statement.

Usage Notes:

- You can create an encrypted password in the PowerExchange Navigator by selecting **File > Encrypt Password**.
- If both of the following conditions apply, PowerExchange uses the default value, which is an empty string, as the search user password:
 - You do not define the LDAP_BIND_PWD statement or the LDAP_BIND_EPWD statement.
 - You do not both define the LDAP_TLS statement to require client certification and define the LDAP_SASL_MECH=EXTERNAL statement.An empty string is acceptable if the LDAP server has been configured to allow anonymous access.
- If you define the LDAP_TLS statement to require client certification and define the LDAP_SASL_MECH=EXTERNAL statement, the value of LDAP_BIND_EPWD is ignored. The credentials of the search user are determined from the client certificate and the certificate mapping of the LDAP server.

RELATED TOPICS:

- [“LDAP Search User” on page 283](#)

LDAP_BIND_PWD Statement

The LDAP_BIND_PWD statement specifies the password for the LDAP search user.

Operating Systems: Linux, UNIX, and Windows

Related Statements: LDAP_BIND_DN, LDAP_BIND_EPWD, LDAP_SASL_MECH

Required: No

Syntax:

```
LDAP_BIND_pwd=search_user_password
```

Value: For the *search_user_password* variable, enter the password for the LDAP search user that is identified in the LDAP_BIND_DN statement.

Usage Notes:

- If both of the following conditions apply, PowerExchange uses the default value, which is an empty string, as the search user password:
 - You do not define the LDAP_BIND_PWD statement or the LDAP_BIND_EPWD statement.
 - You do not both define the LDAP_TLS statement to require client certification and define the LDAP_SASL_MECH=EXTERNAL statement.

An empty string is acceptable if the LDAP server has been configured to allow anonymous access.

- If you define the LDAP_TLS statement to require client certification and define the LDAP_SASL_MECH=EXTERNAL statement, the value of LDAP_BIND_PWD is ignored. The credentials of the search user are determined from the client certificate and the certificate mapping of the LDAP server.

RELATED TOPICS:

- [“LDAP Search User” on page 283](#)

LDAP_BIND_TIMEOUT Statement

The LDAP_BIND_TIMEOUT statement specifies the number of seconds that a PowerExchange Listener or PowerExchange Logger for Linux, UNIX, and Windows waits for an LDAP bind request.

Operating Systems: Linux, UNIX, and Windows

Related Statements: LDAP_BIND_DN, LDAP_TLS

Required: No

Syntax:

```
LDAP_BIND_TIMEOUT={bind_timeout_seconds|120}
```

Value: For the *bind_timeout_seconds* variable, enter a number from 1 to 60000. Default is 120.

Usage Notes:

- The default timeout period is 120 seconds. If this period expires, the bind times out. A signon failure occurs because the PowerExchange user cannot be authenticated.

- LDAP binds occur twice during PowerExchange user authentication: once to authenticate the search user and a second time to authenticate the PowerExchange user. Each bind uses the LDAP_BIND_TIMEOUT value.

RELATED TOPICS:

- [“Controlling the LDAP Search” on page 283](#)

LDAP_FILTER Statement

The LDAP_FILTER statement specifies a filter to speed up or restrict the LDAP search for a PowerExchange user.

Operating Systems: Linux, UNIX, and Windows

Related Statements: LDAP_BASE, LDAP_LOGIN_ATTRIBUTE, LDAP_SCOPE

Required: No

Syntax:

```
LDAP_FILTER="filter"
```

Value: For the *filter* variable, enter a valid LDAP search filter, as specified in the Internet Engineering Task Force RFC 4515 publication at <http://tools.ietf.org/search/rfc4515>. Enclose the search filter in double quotation marks ("").

If you do not include this statement, PowerExchange uses the default value of "(objectclass=*)".

You can validate a filter for correctness by using the ldapsearch command line utility.

RELATED TOPICS:

- [“Controlling the LDAP Search” on page 283](#)

LDAP_HOST Statement

The LDAP_HOST statement defines TCP/IP host details for an LDAP server that PowerExchange might use for user validation. To configure LDAP for failover, define multiple LDAP_HOST statements.

If you specify LDAP as the third parameter of the SECURITY statement, include at least one LDAP_HOST statement in the DBMOVER configuration file. You can include up to 32 LDAP_HOST statements. If you include multiple LDAP_HOST statements, PowerExchange uses the first available LDAP server, based on the order in which the LDAP_HOST statements occur in the DBMOVER configuration file.

Operating Systems: Linux, UNIX, and Windows

Related Statements: LDAP_PORT

Required: No

Syntax:

```
LDAP_HOST=hostname [,port]
```

Parameters:

host_name

Required. TCP/IP host name or IP address of an LDAP server that is listening on the port specified in the *port* parameter or LDAP_PORT statement.

port

Optional. TCP/IP port on which the LDAP server that is specified in the *host_name* parameter is listening. Valid values are from 1 through 65535. If you do not specify the *port* value, PowerExchange uses the value specified in the LDAP_PORT statement. If you do not specify the LDAP_PORT statement either, PowerExchange uses the default of 389.

Usage Notes:

- If you specify multiple LDAP_HOST statements, PowerExchange checks for the first available LDAP server each time a PowerExchange user requires LDAP authentication. If a failed LDAP server becomes available later, it will be included in the checking again automatically.
- Order LDAP_HOST statements in the DBMOVER configuration file to reflect how you prioritize your LDAP servers for PowerExchange authentication use.

RELATED TOPICS:

- [“LDAP Server Failover” on page 285](#)

LDAP_LOGIN_ATTRIBUTE Statement

The LDAP_LOGIN_ATTRIBUTE statement specifies an LDAP attribute to use as the search key to match against a PowerExchange user ID.

Operating Systems: Linux, UNIX, and Windows

Related Statements: LDAP_BASE, LDAP_FILTER, LDAP_SCOPE

Required: No

Syntax:

```
LDAP_LOGIN_ATTRIBUTE=login_attribute
```

Value: For the *login_attribute* variable, enter the LDAP attribute to use as the search key to match against a PowerExchange user ID.

If you do not include this statement, PowerExchange uses the default value of "uid". This value causes the PowerExchange user search to be for an LDAP entry that contains a uid attribute that matches the PowerExchange user ID.

RELATED TOPICS:

- [“Controlling the LDAP Search” on page 283](#)

LDAP_OPENSSL Statement

The LDAP_OPENSSL statement enables LDAP secured connections between PowerExchange and the LDAP server and specifies certificate information for a Transport Layer Security (TLS) connection to the LDAP server. You can also configure PowerExchange to use the StartTLS extension to initiate LDAP secured communications.

If you are using an OpenLDAP client, define the LDAP_OPENSSL statement and specify OPEN_LDAP for the fourth positional parameter in the SECURITY statement.

If you are using an Oracle LDAP client, use the LDAP_TLS statement instead of the LDAP_OPENSSL statement.

Operating Systems: Linux, UNIX, and Windows

Related Statements: LDAP_SASL_MECH, LDAP_TLS

Required: No

Syntax:

```
LDAP_OPENSSL=({CAPATH=directory|CAFILE=filepath}  
[ ,CERTFILE=filepath,KEYFILE=filepath]  
[ ,PASS=passphrase|EPASS=encrypted_passphrase]  
[ ,START_TLS={N|Y}]  
)
```

Parameters:

CAPATH=*directory*

Required if CAFILE is not specified. Directory where OpenSSL can find CA certificate files in PEM format.

CAFILE=*filepath*

Required if CAPATH is not specified. File that contains one or more CA certificates in PEM format.

CERTFILE=*filepath*

Optional. Client signing certificate. Include this parameter if the LDAP server is configured to require a signed certificate from its clients. The certificate and key files must be in PEM format. The certificate file must be named by the hash of the CA certificate.

KEYFILE=*filepath*

Required if CERTFILE is specified. Client private key for signing its certificate. The certificate and key files must be in PEM format.

PASS=*passphrase*

Optional. If the key file is DES-encrypted, the passphrase that is used to access the private key that is associated with the client certificate. Do not enter both the PASS and the EPASS parameters.

EPASS=*encrypted_passphrase*

Optional. If the key file is DES-encrypted, the encrypted passphrase that is used to access the private key that is associated with the client certificate. Do not enter both the PASS and the EPASS parameters.

START_TLS={N|Y}

Optional. Controls whether PowerExchange uses the StartTLS extended LDAP operation to initiate secure network traffic on a normally unsecured port. Default is N.

RELATED TOPICS:

- [“Levels of LDAP Security” on page 284](#)
- [“LDAP StartTLS Extended Operation” on page 285](#)

LDAP_PORT Statement

The LDAP_PORT statement defines the TCP/IP port for the LDAP servers that you defined in the LDAP_HOST statement and that can be used for PowerExchange user authentication.

Operating Systems: Linux, UNIX, and Windows

Related Statements: LDAP_HOST, LDAP_TLS

Required: No

Syntax:

```
LDAP_PORT={port|389}
```

Value: For the *port* variable, enter the TCP/IP port on which your LDAP servers are listening. Valid values are from 1 through 65535. Default is 389.

Usage Notes:

- The *port* value that you specify applies to all LDAP servers that you define in LDAP_HOST statements. To specify a different port for a particular LDAP server, include the *port* parameter in the LDAP_HOST statement.

LDAP_SASL_MECH Statement

The LDAP_SASL_MECH statement specifies the authentication mechanism that the Simple Authentication and Security Layer (SASL) uses. You can define this statement if you define the LDAP_TLS statement or the LDAP_OPENSSL statement to require client certification.

Operating Systems: Linux, UNIX, and Windows

Related Statements: LDAP_BIND_DN, LDAP_TLS, LDAP_OPENSSL

Required: No

Syntax:

```
LDAP_SASL_MECH=SASL_mechanism
```

Value: For the *SASL_mechanism* variable, specify EXTERNAL. PowerExchange uses the SASL EXTERNAL authentication mechanism with LDAP. With this option, you do not need to specify the identity of the LDAP search user by including the LDAP_BIND_DN statement. Instead, the LDAP search user identity is determined from a TLS client certificate, together with the certificate mapping configuration of the LDAP server.

Usage Notes:

- If you do not define the LDAP_BIND_DN statement or the LDAP_SASL_MECH statement, the LDAP client does not attempt to perform a bind and instead performs unauthenticated operations.

RELATED TOPICS:

- [“Levels of LDAP Security” on page 284](#)

LDAP_SCOPE Statement

The LDAP_SCOPE statement defines the search depth in the LDAP tree for LDAP searches.

Operating Systems: Linux, UNIX, and Windows

Related Statements: LDAP_BASE, LDAP_FILTER, LDAP_LOGIN_ATTRIBUTE

Required: No

Syntax:

```
LDAP_SCOPE={BASE|ONE|SUBTREE}
```

Valid Values:

- **BASE.** PowerExchange searches only the entry at the base DN. As a result, only that entry is returned.
- **ONE.** PowerExchange searches all entries one level under the base DN, not including the base DN and not including subordinate entries.

- **SUBTREE.** PowerExchange searches the entry at the base DN and all entries at all levels under the base DN.

Default is SUBTREE.

Usage Notes:

- The LDAP_SCOPE statement applies even if you define multiple LDAP_BASE statements. In this case, each base is searched using the same value that is specified for LDAP_SCOPE.

RELATED TOPICS:

- [“Controlling the LDAP Search” on page 283](#)

LDAP_SEARCH_TIMEOUT Statement

The LDAP_SEARCH_TIMEOUT statement specifies the number of seconds that a PowerExchange Listener or PowerExchange Logger for Linux, UNIX, and Windows waits to receive a search result during PowerExchange user validation against LDAP.

Operating Systems: Linux, UNIX, and Windows

Related Statements: LDAP_BIND_DN, LDAP_TLS

Required: No

Syntax:

```
LDAP_SEARCH_TIMEOUT={seconds|120}
```

Value: For the *seconds* variable, enter the number of seconds that a PowerExchange Listener or PowerExchange Logger for Linux, UNIX, and Windows waits to receive a search result during PowerExchange user validation against LDAP. Valid values are 1 to 60000. Default is 120.

Usage Notes:

- The default timeout period is 120 seconds. If this period expires, the search times out. A signon failure occurs because the PowerExchange user cannot be authenticated.

RELATED TOPICS:

- [“Controlling the LDAP Search” on page 283](#)

LDAP_TLS Statement

The LDAP_TLS statement enables LDAP secured connections between PowerExchange and the LDAP server and specifies certificate information for a Transport Layer Security (TLS) connection to the LDAP server. You can also configure PowerExchange to use the StartTLS extension to initiate LDAP secured communications.

If you are using an Oracle LDAP client, define the LDAP_TLS statement, and use the default value of ORACLE_LDAP for the fourth positional parameter in the SECURITY statement.

If you are using an OpenLDAP client, use the LDAP_OPENSSL statement instead of the LDAP_TLS statement.

Operating Systems: Linux and UNIX

Related Statements: LDAP_SASL_MECH, LDAP_OPENSSL

Required: No

Syntax:

```
LDAP_TLS=({PASS=client_passphrase|EPASS=client_encrypted_passphrase}
           [,KEYNAME=key_name]
           [,CAPATH=directory]
           [,START_TLS=Y|N]
        )
```

Parameters:**PASS=*client***

Optional. The passphrase that is used to access the private key that is associated with the client certificate. Do not enter both the PASS and the EPASS parameters.

EPASS=*client_encrypted*

Optional. The encrypted passphrase that is used to access the private key that is associated with the client certificate. Do not enter both the PASS and the EPASS parameters.

Tip: You can create an encrypted passphrase in the PowerExchange Navigator by selecting **File > Encrypt Password**.

KEYNAME=*key_name*

Optional. The nickname of the client certificate that is used to make an LDAP TLS connection.

CAPATH=*directory*

Optional. The directory that is used for the client keystore and CA truststore.

START_TLS={Y|N}

Optional. Controls whether PowerExchange uses the StartTLS extended LDAP operation to initiate secure network traffic on a normally unsecured port. Default is N.

RELATED TOPICS:

- [“Levels of LDAP Security” on page 284](#)
- [“LDAP StartTLS Extended Operation” on page 285](#)

LISTENER Statement

The LISTENER statement defines the TCP/IP port on which a named PowerExchange Listener process listens for work requests.

You can define up to 10 LISTENER statements in a DBMOVE configuration file.

For netport jobs on z/OS, define a LISTENER statement with a unique port and define a NETPORT statement that references that port.

Optionally, you can specify SSL authentication and additional parameters that control TCP/IP buffer sizes and wait times.

Operating Systems: All

Related Statements: NETPORT for netport jobs and SSL for SSL authentication

Required: No

Syntax:

```
LISTENER=({listener_node|node1}
           ,TCP/IP
           ,{port|2480})
```

```

    [, {send_bufsize|65536}]
    [, {receive_bufsize|65536}]
    [, {send_size|4096}]
    [, {receive_size|4096}]
    [, receive_timeout]
    [, ip_address]
    [, SSL]
)

```

Parameters:

{listener_node|node1}

Required. Node name of the TCP/IP port on which the PowerExchange Listener process listens.

Use this node name to select a LISTENER statement when you start the PowerExchange Listener process, as follows:

- On Linux, UNIX, and Windows, specify the listener node name on the dtllst command.
- On i5/OS, specify the listener node name in the SBMJOB command that runs the DTLLST program. For more information, see the *PowerExchange Command Reference*.
- On z/OS, specify the listener node name in the PARM field of the EXEC card in the JCL.

Default is node1.

TCPIP

Required. Communications protocol. TCPIP is the only valid option.

{port|2480}

Required. TCP/IP port used to listen for work requests. Valid values are from 1 through 65535. Default is 2480.

{send_bufsize|65536}

Optional. Size, in bytes, of the data portion of the TCP/IP send buffer. Valid values are from 1024 through 1048576. Default is 65536.

{receive_bufsize|65536}

Optional. Size, in bytes, of the data portion of the TCP/IP receive buffer. Valid values are from 1024 through 1048576. Default is 65536.

{send_size|4096}

Optional. Maximum size, in bytes, of the block of data that PowerExchange sends to TCP/IP at one time. If the data exceeds this size, PowerExchange splits data into multiple blocks until all of the data is sent. Valid values are from 512 through 1048576. Default is 4096.

Tip: Enter a value that is less than or equal to the TCP/IP send buffer size.

{receive_size|4096}

Optional. Maximum size, in bytes, of the block of data that PowerExchange processes from TCP/IP in one operation. If the data exceeds this size, PowerExchange splits data into multiple blocks until all of the data is received. Valid values are from 512 through 1048576. Default is 4096.

Tip: Enter a value that is greater than or equal to the TCP/IP receive buffer size.

receive_timeout

Optional. Number of seconds that PowerExchange uses as the receive timeout value when a long wait is required. PowerExchange uses this value for this request only. Valid values are from 1 through 14400.

ip_address

Optional. IP address that PowerExchange uses on the bind operation for the socket.

If you do not specify an IP address, PowerExchange uses INADDR_ANY on the bind operation, which causes TCP/IP to bind to all network interfaces on the host. Use this parameter if you have multiple network interfaces and want to restrict the port on which the PowerExchange Listener listens to a specific interface.

SSL

Optional. Specifies that PowerExchange uses SSL authentication. Specify this parameter on a Linux, UNIX, or Windows Listener machine that serves as the SSL server during SSL communications.

Usage Notes:

- If you create a PowerExchange Listener Service, use the node name that you define in the LISTENER statement as follows:
 - If you create the Listener Service through Informatica Administrator, the node name value that you specify in the **Start Parameters** property must match the node name that you define in the LISTENER statement.
 - If you create the Listener Service through the infacmd pwx CreateListenerService command, the node name value that you specify for the -StartParameters option on the command must match the node name that you define in the LISTENER statement.

When you create the Listener Service, the Service Manager associates it with the PowerExchange Listener process on the node. For more information about configuring and creating a Listener Service, see the *Informatica Application Service Guide*.

- If you issue pwxcmd commands or infacmd pwx commands to the PowerExchange Listener process, the node name that you define in the LISTENER statement must match the node name that you define in the SVCNODE statement in the DBMOVER configuration file. In this case, enter a string of up to 12 characters in length for the *listener_node* parameter of the LISTENER statement. This is the maximum length of the service name that you can specify in the SVCNODE statement.

LOADCTLFILE Statement

The LOADCTLFILE statement specifies the PDS data set that contains the control card template member for DB2 for z/OS LOAD utility batch jobs.

Operating Systems: z/OS

Data Sources: DB2 for z/OS

Required: No

Syntax:

```
LOADCTLFILE={pds_name|A}
```

Value: For the *pds_name* variable, enter the PDS data set that contains the control card template member for DB2 for z/OS LOAD utility batch jobs. PowerExchange reads this data set on the system where you perform the bulk load. Default is A.

Usage Notes:

- When you install PowerExchange, the z/OS Installation Assistant includes your RUNLIB data set name in the LOADCTLFILE statement in the DBMOVER member.

- PowerExchange provides the following DB2 LOAD control card template members in RUNLIB:
 - **DB2LDCTL**. Sample control card statements for nonpartitioned tables.
 - **DB2LDCTP**. Sample control card statements for partitioned tables.

LOADJOBFILE Statement

The LOADJOBFILE statement specifies the PDS data set that contains the JCL template member for DB2 for z/OS LOAD utility and CA IDMS/DB metadata retrieval batch jobs.

Operating Systems: z/OS

Data Sources: CA IDMS/DB and DB2 for z/OS

Related Statements: SUBMITTIMEOUT

Required: No

Syntax:

```
LOADJOBFILE={pds_name|A}
```

Value: For the *pds_name* variable, enter the PDS data set that contains the JCL template member for DB2 for z/OS LOAD utility and CA IDMS/DB metadata retrieval batch jobs. For DB2 LOAD utility operations, PowerExchange reads this data set on the system where you perform the bulk load. Default is A.

Usage Notes:

- When you install PowerExchange, the z/OS Installation Assistant includes your RUNLIB data set name in the LOADJOBFILE statement in the DBMOVER member.
- PowerExchange provides the following JCL template members in RUNLIB:
 - **DB2LDJCL**. Sample JCL for DB2 LOAD utility jobs for nonpartitioned tables.
 - **DB2LDJCP**. Sample JCL for DB2 LOAD utility jobs for partitioned tables.
 - **IDMSMJCL**. Sample JCL for CA IDMS/DB metadata retrieval.
 - **IDMSMJCX**. Sample JCL for CA IDMS/DB metadata retrieval that creates a temporary load library for the subschema load module.
- By default, a PowerExchange Listener waits for 60 seconds for the spawned jobs to start. You can increase this timeout period by defining the SUBMITTIMEOUT statement. If the batch job does not start in the timeout period, PowerExchange times out the job, stops the task in the PowerExchange Listener, and writes the PWX-00426 message to the PowerExchange message log.

LOG_CODEPAGE Statement

The LOG_CODEPAGE statement specifies the code page that PowerExchange uses to write messages to the log file.

On Windows, programs read log records correctly if they use the same code page. Define this statement to direct PowerExchange to use a different code page than the control code page to write messages to the log file.

Operating Systems: Windows

Required: No

Syntax:

```
LOG_CODEPAGE={code_page|UTF-8}
```

Value: For the *code_page* variable, enter a code page value. Default is UTF-8.

Example: To use the Japanese code page to write messages to the log file, define the following statement:

```
LOG_CODEPAGE=CP943
```

LOG_LINE_LIMIT Statement

The LOG_LINE_LIMIT statement specifies the maximum line length for PowerExchange messages in the message log.

Operating Systems: i5/OS, Linux, UNIX, Windows, and z/OS

Data Sources: Not applicable

Required: No

Syntax:

```
LOG_LINE_LIMIT={number_of_characters|79}
```

Value: For the *number_of_characters* variable, enter the number of characters that defines the maximum line length for messages in the message log. Messages that span multiple lines, such as messages that report diagnostic output or statistics messages in tabular format, might be improperly formatted if the log line limit is too short. In this case, try increasing this value to make the messages easier to read in the log.

Valid values:

- On i5/OS, Linux, UNIX, or Windows, enter a number from 79 to 255.
- On z/OS, enter a number from 79 to 132.

Default is 79 on all of these operating systems.

Usage Notes:

- If you enter a value greater than 80 and do not specify the RECLLEN parameter in the TRACING statement, PowerExchange uses the LOG_LINE_LIMIT value as the RECLLEN value.
- If you enter the MONITOR parameter in the STATS statement to produce summary statistics for a PowerExchange Listener on z/OS, set the LOG_LINE_LIMIT value to 132. Otherwise, the lines in the PowerExchange Listener displaystats report might wrap awkwardly on z/OS, making the report output hard to read.

LOGPATH Statement

The LOGPATH statement specifies a unique path and directory for PowerExchange message log files on a Linux, UNIX, or Windows system.

Define this statement to create message log files in a directory that is separate from your current working directory so that you can find the message log files more easily.

Operating Systems: Linux, UNIX, and Windows

Required: No

Syntax:

```
LOGPATH=directory
```

Value: For the *directory* variable, enter the full path to the directory where you want PowerExchange to write message log files. Default is the current working directory.

Usage Notes: If you also specify a value in the DETAIL_LOGPATH environment variable, the environment variable overrides the LOGPATH statement.

LOGSID Statement

The LOGSID statement specifies the location of the CA IDMS/DB logs and the PowerExchange log catalog.

Define this statement in the DBMOVER configuration file on the system where the catalog resides. For more information, see the *PowerExchange CDC Guide for z/OS*.

Operating Systems: z/OS

Data Sources: CA IDMS/DB

Required: Yes for CA IDMS/DB sources

Syntax:

```
LOGSID=(registration_logsid
        ,listener_node
        ,log_catalog_name
        ,log_catalog_instance_name
      )
```

Parameters:

registration_logsid

Required. The registration logsid, which matches the logsid value specified in the **Add Registration Group** dialog box when you added the registration group.

listener_node

Required. The node of the PowerExchange Listener, which matches the node specified in the LISTENER statement in the DBMOVER configuration file.

log_catalog_name

Required. The name of the PowerExchange log catalog data set.

log_catalog_instance

Required. The log catalog instance name, which matches the value specified for the log entry in the DTLULCAT utility, which creates the input for DTLULOGC.

LOWVALUES Statement

The LOWVALUES statement specifies whether PowerExchange preserves hexadecimal '0' values, called *low values*, in source character fields when passing these values to a PowerCenter session.

When the session runs, the PowerExchange Client for PowerCenter (PWXPC) can write these values to the target. The LOWVALUES statement applies to PowerExchange NRDB, DB2, and CDC sources and targets. If you use PowerExchange with Informatica Developer, the LOWVALUES statement has no effect.

Operating Systems: Linux, UNIX, Windows, z/OS

Required: No

Syntax:

```
LOWVALUES={N|Y}
```

Valid Values:

- **N.** PowerExchange does not preserve hexadecimal '0' values. It interprets a hexadecimal '0' value as the end of a column string and pads the remaining length of the string with spaces. The portion of the column string after the hexadecimal '0' value is not written to the target.
- **Y.** PowerExchange preserves hexadecimal '0' values in the column string. The entire column string, including the low value, is written to the target.

Default is N.

Usage Notes:

- To ensure that the source hexadecimal '0' values are preserved during a PowerCenter session, you must use the PowerExchange Client for PowerCenter (PWXPC) interface instead of the PowerExchange ODBC interface.
- To write hexadecimal '0' values from a supported data source to a supported data target without any translation of the values, complete the following tasks:
 - Specify LOWVALUES=Y in the DBMOVER configuration file on the Integration Service machine.
 - In Powercenter Workflow Manager, on the **Config Object** tab for the session, enter `PreserveLowValues=Yes` in the **Custom Properties** field.
 - In the source and target connections in the PowerCenter workflow, ensure that the **Convert character data to string** option is *not* selected.
- In PowerCenter, you can override the LOWVALUES setting in the DBMOVER configuration file by entering LOWVALUES={Y|N} in the **PWX Override** connection attribute. You can then use the connection in a session to implement the override for a specific source or access method.

LRECL Statement

The LRECL statement specifies the logical record length that PowerExchange uses when dynamically allocating target data sets on z/OS, if you do not provide LRECL information.

Operating Systems: z/OS

Related Statements: BS, DISP, RECFM, RELEASE, SPACE, UNIT, and VOLSER

Required: No

Syntax:

```
LRECL=record_length
```

Value: For the *record_length* value, enter a number from 4 through 32756. Default is 32752.

Usage Notes: If you omit both the LRECL and RECFM statements, PowerExchange uses 32752 for the LRECL value and VB for the RECFM value to allocate data sets.

LU00FILE Statement

The LU00FILE statement specifies the partitioned data set extended (PDSE) that you allocated to hold database description (DBD) output from the IMS Catalog Library Builder Utility, DFS3LU00. This statement is required to retrieve DBD information in source format from the IMS catalog for use in creating data maps.

PowerExchange can transparently use the DFS3LU00 utility to extract DBDs in source format from the IMS catalog, write the DBD information to a pre-allocated PDSE, and transmit the DBD information from the PDSE to the PowerExchange Navigator when you create data maps for source objects.

Operating Systems: z/OS

Data Sources: IMS

Required: Required to use the DFS3LU00 utility to get DBD information from the IMS catalog

Syntax:

```
LU00FILE=pdse_name
```

Parameter:

pdse_name

Required. The name of the PDSE that you allocated to hold the DBD output from the IMS Catalog Library Builder Utility, DFS3LU00.

MAXTASKS Statement

The MAXTASKS statement defines the maximum number of concurrent tasks that can run under the PowerExchange Listener.

Operating Systems: All

Required: No

Syntax:

```
MAXTASKS={maximum_tasks|5}
```

Value: For the *maximum_tasks* variable, enter a number from 1 through 255. Default is 5.

Usage Notes:

- If resource usage by PowerExchange Listener tasks exceeds the resources available for the PowerExchange Listener, the Listener abends. In this case, you can try decreasing the MAXTASKS value to run fewer concurrent tasks. Although tasks can have different resource requirements, this statement can help limit the amount of resources that are used for PowerExchange Listener tasks and prevent these types of abends.
- The MAXTASKS parameter is not intended to be an operating system performance and tuning parameter. To balance the workload of multiple concurrent tasks, use the workload management facility of the operating system, for example, Workload Management (WLM) on z/OS.
- If you use PowerExchange tuning features that result in multiple threads and PowerExchange Listener subtasks, such as pipeline partitioning or multithreaded processing, you might need to increase the MAXTASKS value if PowerExchange processing slows or hangs. You also might need to increase this value if you add PowerCenter workflows. After the number of concurrent tasks reaches the MAXTASKS limit, the PowerExchange Listener rejects requests for additional tasks with message PWX- 00609:

```
PWX-00650 10.3.0.111:3720 : Listener 10.33.40.42 -> 10.3.0.1 on port 1234 socket 51  
PWX-00609 Listener has temporarily stopped accepting connections.
```

When the number of the concurrent tasks drops below the MAXTASKS limit once again, the PowerExchange Listener begins accepting requests for additional tasks.

- If you use connection pooling, ensure that the MAXTASKS value is large enough to accommodate the size of the connection pool.
- The maximum number of concurrent tasks that a PowerExchange Listener can support might be substantially less than 255 because of operating system resource constraints, such as virtual storage limitations on z/OS.

MSSQL_SERVER_CONNECT_TIMEOUT Statement

The MSSQL_SERVER_CONNECT_TIMEOUT statement specifies the timeout interval, in seconds, for a PowerExchange Navigator, DTLUCBRG utility, or DTLURDMO utility connection to a Microsoft SQL Server Management Objects (SMO) interface to manage SQL Server publications.

After this interval elapses, the connection times out with error message PWX-15700. If you receive PWX-15700 messages for a timeout error, use this statement to increase the timeout interval.

Operating Systems: Windows

Data Sources: Microsoft SQL Server

Related Statements: MSSQL_SERVER_STATEMENT_TIMEOUT

Required: No

Syntax:

```
MSSQL_SERVER_CONNECT_TIMEOUT={seconds|60}
```

Value: For the *seconds* value, enter a number from 1 to 86400. Default is 60 seconds.

MSSQL_SERVER_STATEMENT_TIMEOUT Statement

The MSSQL_SERVER_STATEMENT_TIMEOUT statement specifies the timeout interval, in seconds, for processing a Transact-SQL statement issued by the PowerExchange Navigator, DTLUCBRG utility, or DTLURDMO utility to the Microsoft SQL Server Management Objects (SMO) interface.

After this interval elapses, processing of the Transact-SQL statement stops with error message PWX-15700. If you receive PWX-15700 messages related to this error, use this statement to increase the timeout interval.

Operating Systems: Windows

Data Sources: Microsoft SQL Server

Related Statements: MSSQL_SERVER_CONNECT_TIMEOUT

Required: No

Syntax:

```
MSSQL_SERVER_STATEMENT_TIMEOUT={seconds|60}
```

Value: For the *seconds* value, enter a number from 1 to 86400. Default is 60 seconds.

MSGPREFIX Statement

The MSGPREFIX statement defines the string that PowerExchange uses to prefix PowerExchange messages.

Operating Systems: All

Related Statements: MSGPREFIX-HYPHEN

Required: No

Syntax:

```
MSGPREFIX={prefix|PWX}
```

Value: For the *prefix* variable, enter a one- to eight-character alphanumeric string. Default is PWX.

MSGPREFIX-HYPHEN Statement

The MSGPREFIX-HYPHEN statement specifies whether PowerExchange includes or excludes the hyphen character between the message prefix and the message number.

Operating Systems: All

Related Statements: MSGPREFIX

Required: No

Syntax:

```
MSGPREFIX-HYPHEN={N|Y}
```

Valid Values:

- **N.** PowerExchange excludes the hyphen character between the message prefix and the message number.
- **Y.** PowerExchange includes the hyphen character between the message prefix and the message number.

Default is Y.

Examples:

- To include the hyphen character between the message prefix and message number, accept the default or define the following statement:

```
MSGPREFIX-HYPHEN=Y
```

For the 06111 message, PowerExchange issues PWX-06111.

- To exclude the hyphen character between the message prefix and message number, define the following statement:

```
MSGPREFIX-HYPHEN=N
```

For the 06111 message, PowerExchange issues PWX06111.

MSS_ERRORFILE Statement

The MSS_ERRORFILE statement specifies the name of the user-customized SQL error file that PowerExchange uses for Microsoft SQL Server bulk data movement operations.

Operating Systems: Windows

Data Sources: Microsoft SQL Server

Required: No

Syntax:

```
MSS_ERRORFILE=file_name
```

Value: For the *file_name* variable, enter the complete path and file name that contains the SQL error codes that you want PowerExchange to treat as either recoverable or fatal. PowerExchange provides a sample error action file called mssqlerr.act in the PowerExchange installation directory.

MVSDB2AF Statement

The MVSDB2AF statement specifies which DB2 for z/OS attachment facility PowerExchange uses for DB2 bulk data movement operations.

Operating Systems: z/OS

Data Sources: DB2 for z/OS

Required: No

Syntax:

```
MVSDB2AF={CAF|RRSAF}
```

Valid Values:

- **CAF.** PowerExchange uses the DB2 call attachment facility (CAF) to connect to DB2.

When you use CAF, PowerExchange and DB2 use the user ID of the PowerExchange Listener or netport job for authentication of DB2 resources.

- **RRSAF.** PowerExchange uses the DB2 Resource Recovery Services attachment facility (RRSAF) to connect to DB2. To control user access to DB2 tables, specify RRSAF.

RRSAF enables PowerExchange to sign on to DB2 and use the user ID of the user that requested the bulk data movement operation for authorization of DB2 resources.

To use RRSAF, you must configure and run RRS on the z/OS system.

Default is CAF.

Usage Notes: If you specify RRSAF, PowerExchange uses the following values for correlation-id and accounting-token when signing on to RRSAF:

correlation-id

PowerExchange uses one of the following values:

- If provided, PowerExchange uses the value specified for the Correlation ID connection property in the PowerCenter Workflow Manager or informatica Developer tool.
- If no value is specified in the client for the Correlation ID connection property, PowerExchange uses the value specified in the SESSID statement in the DBMOVER configuration file on the PowerExchange Listener machine.
- If no value is specified in the client Correlation ID connection property or in the DBMOVER configuration file on the PowerExchange Listener machine, PowerExchange uses the default value of DETAIL.

accounting-token

PowerExchange concatenates the 8-byte job name, 8-byte user ID, and 6-byte task ID:

```
jobname||user_id||task_id
```

RELATED TOPICS:

- [“DB2 for z/OS Security” on page 276](#)

NEGSIGN Statement

The NEGSIGN statement defines the character that PowerExchange uses as the negative sign character in fields that contain negative numbers.

Data Sources: All

Required: No

Syntax:

```
NEGSIGN={character|_}
```

Value: For the *character* variable, enter a single character. Default is the dash (-) character.

NETPORT Statement

The NETPORT statement defines information for a netport job and associates the netport job with a specific listener port. You must specify a corresponding LISTENER statement.

Operating Systems: z/OS

Data Sources: All

Related Statements: ADAUSER, GDGLOCATE, LISTENER, SUBMITTIMEOUT, TAPEWAIT, and WAITDSN

Required: No

Syntax:

```

NETPORT=(listener_node
        ,port
        [,cpnode]
        [,cpport]
        ,netport_member
        [,substitution_1]
        [,substitution_2]
        [,substitution_3]
        [,substitution_4]
        [,substitution_5]
        [,substitution_6]
        [,substitution_7]
        [,substitution_8]
        )

```

Parameters:***listener_node***

Required. Listener node name from the LISTENER statement associated with this NETPORT statement.

This value must also match the node name in the PARM parameter of the EXEC JCL statement for the running PowerExchange Listener. Otherwise, the PowerExchange Listener does not listen on the listener port on the LISTENER statement.

port

Required. TCP/IP port from the LISTENER statement associated with this NETPORT statement. The PowerExchange Listener uses the port on the associated LISTENER statement to listen for netport job requests.

Valid values are from 1 through 65535.

cpnode

Optional. Listener node name in the DBMOVER member that the netport job uses. If the netport JCL executes the DTLLST3 program, the DBMOVER member specified in the DTLCFG DD statement must contain a LISTENER statement with this node name.

Default is the *listener_node* parameter value.

cpport

Optional. TCP/IP port in the DBMOVER member that the netport job uses. If the netport JCL uses the DTLLST3 program, the DBMOVER member specified in the DTLCFG DD statement must contain a LISTENER statement with this port.

Default is the *port* parameter value.

netport_member

Required. Data set that contains the netport JCL. You can use either a sequential data set or a PDS with a member.

Note: If you use a PDS, you must specify the member name in parentheses and enclose the entire value in quotes ("").

substitution_1

Optional. Populates the netport JCL substitution variable %1. In the IMSJCL member of the RUNLIB library, PowerExchange uses this variable to populate the PSB=%1 parameter in the DLIBATCH PROC.

substitution_2

Optional. Netport JCL substitution variable %2.

substitution_3

Optional. Netport JCL substitution variable %3.

substitution_4

Optional. Netport JCL substitution variable %4.

substitution_5

Optional. Netport JCL substitution variable %5.

substitution_6

Optional. Netport JCL substitution variable %6.

substitution_7

Optional. Netport JCL substitution variable %7.

substitution_8

Optional. Netport JCL substitution variable %8.

Usage Notes:

- You can enter up to ten NETPORT statements in the DBMOVER file.
- PowerExchange uses the substitution parameter values in the NETPORT statement, if specified, to replace substitution variables in the netport JCL before submitting the JCL.
- Use a netport job to invoke the following programs to complete PowerExchange and non-PowerExchange functions:
 - The IMS DLIBATCH PROC or DFSRR00 program to access IMS databases.
 - The batch PowerExchange Listener program, DTLLST3, to access data sets.

With DTLLST3, the netport job can wait for tape mounts or data sets without impacting other users of the PowerExchange Listener.

 - The batch PowerExchange Listener program, DTLLST3, to access CA IDMS/DB tables with full user ID checking.
- By default, a PowerExchange Listener waits 60 seconds for spawned jobs to start. To increase this timeout period, define a SUBMITTIMEOUT statement. If the batch job does not start in the timeout period, PowerExchange times out the job, stops the task in the PowerExchange Listener, and writes the PWX-00426 message to the PowerExchange message log.
- For IMS sources and targets, the netport job can access an IMS source or target database for bulk data movement. Define this statement only if you use the DL/1 BATCH access method, which provides DL/I or BMP access to the IMS database. A NETPORT statement is not required for ODBA access to IMS data.

You can override the PSB name in the NETPORT statement, if specified, in the PowerCenter session properties for the source or target and in the PowerExchange Navigator advanced parameters for a database row test. In this case, you must include the %PSBNAME substitution variable in the netport JCL. You can use the overrides to prevent exceeding the limit of ten NETPORT statements in the DBMOVER file.

If you include the PSB=%1 substitution variable in the netport JCL, the PSB name is taken from the sixth parameter (*substitution_1*) in the NETPORT statement.

Example: The following example shows sample LISTENER and NETPORT statements that use the IMSJCL member from the RUNLIB library:

```
LISTENER=(node1,TCPIP,9999)
NETPORT=(node1,9999,,,"PWX.RUNLIB(IMSJCL)",myspb,sub2,sub3,sub4)
```

This sample NETPORT statement passes a PSB called mypsb to the IMSJCL member. This statement also passes three additional values to use as substitution variables in the IMSJCL member.

RELATED TOPICS:

- [“Netport Jobs” on page 209](#)

NODE Statement

The NODE statement defines the TCP/IP host name and port that PowerExchange uses to contact a PowerExchange Listener process.

You can specify up to 128 NODE statements in a DBMOVER configuration file.

Optionally, specify SSL authentication and additional parameters that control TCP/IP buffer sizes and wait times. And optionally, specify the *service_name* parameter to identify a PowerExchange Listener Service.

Operating Systems: All

Related Statements: NETPORT for netport jobs and SSL for SSL authentication

Required: No

Syntax:

```
NODE=({node_name|node1}
      ,TCPIP
      ,host_name
      ,{port|2480}
      [, {send_bufsize|65536}]
      [, {receive_bufsize|65536}]
      [, {send_size|4096}]
      [, {receive_size|4096}]
      [, receive_timeout]
      [, {SSL|ZOSSL}]
      [, service_name]
      )
```

Parameters:

node_name|node1

Required. Unique, user-defined name for this NODE statement. The name does not need to match the name of the PowerExchange Listener process. To contact the PowerExchange Listener process to which the statement points, enter this name in user interfaces that prompt for the location of the PowerExchange Listener, including the following interfaces:

- The **Location** attribute in a connection definition in the Informatica Developer or PowerCenter Workflow Manager
- The **Location** attribute in a source or target definition in the PowerCenter Designer
- The **Location** field in the PowerExchange Navigator dialog boxes

Default is node1.

TCPIP

Required. Communications protocol. TCPIP is the only valid option.

host_name

Required. TCP/IP host name or IP address for the PowerExchange Listener process that listens on the port specified in the *port* parameter. If the *service_name* parameter is specified, *host_name* is ignored.

{port|2480}

Required. TCP/IP port on which the PowerExchange Listener process that runs on the system specified in *host_name* listens. Valid values are from 1 through 65535. Default is 2480.

{send_bufsize|65536}

Optional. Size, in bytes, of the data portion of the TCP/IP send buffer. Valid values are from 1024 through 1048576. Default is 65536.

{receive_bufsize|65536}

Optional. Size, in bytes, of the data portion of the TCP/IP receive buffer. Valid values are from 1024 through 1048576. Default is 65536.

{send_size|4096}

Optional. Maximum size, in bytes, of the block of data that PowerExchange sends to TCP/IP at one time. If the data exceeds this size, PowerExchange splits data into multiple blocks until it sends all the data. Valid values are from 512 through 1048576. Default is 4096.

Tip: Enter a value that is less than or equal to the TCP/IP send buffer size.

{receive_size|4096}

Optional. Maximum size, in bytes, of the block of data that PowerExchange processes from TCP/IP in one operation. If the data exceeds this size, PowerExchange splits data into multiple blocks until it receives all the data. Valid values are from 512 through 1048576. Default is 4096.

Tip: Enter a value that is greater than or equal to the TCP/IP receive buffer size.

receive_timeout

Optional. Number of seconds that PowerExchange uses as the receive timeout value when a long wait is required. PowerExchange uses this value for this request only. Valid values are from 1 through 14400.

{SSL|ZOSSL}

Optional. Specifies that PowerExchange uses SSL authentication.

Specify the ZOSSL option only on a Linux, UNIX, or Windows SSL client that is communicating with a z/OS system. Otherwise, specify the SSL option on the Linux, UNIX, or Windows SSL client.

service_name

Optional. To configure an Informatica client tool or integration service to locate a PowerExchange Listener Service in the Informatica domain, specify the name of the Listener Service in the *service_name* parameter.

A client tool is the Developer tool or PowerCenter Client. An integration service is the PowerCenter Integration Service or Data Integration Service.

If you include this parameter, the Informatica client tool or integration service ignores the *host_name* parameter on the NODE statement and uses the *service_name* and *port* parameters to locate the Listener Service in the Informatica domain.

For more information about the PowerExchange Listener Service, see the *Informatica Application Service Guide*.

NOGETHOSTBYNAME Statement

The NOGETHOSTBYNAME statement controls whether PowerExchange gets the IP address of the local system to verify the license and include it in PowerExchange messages.

Operating Systems: All

Required: No

Syntax:

```
NOGETHOSTBYNAME={N|Y}
```

Valid Values:

- **N.** PowerExchange gets the IP address of the local system to verify the PowerExchange license. PowerExchange also includes the IP address in messages written to the PowerExchange message log file, such as message PWX-00651.
- **Y.** PowerExchange does not get the IP address of the local system. Use this option at the direction of Informatica Global Customer Support as it also requires use of a special license key.

Default is N.

Usage Notes: In some situations, resolution of the IP address can take a long time, resulting in poor performance. This situation normally occurs because there is no entry in the TCP/IP HOSTS table for the local system. Define the NOGETHOSTBYNAME statement to temporarily change PowerExchange behavior until you resolve the TCP/IP problem.

NRDB_WRITE_CHAR_NULL_FILL Statement

The NRDB_WRITE_CHAR_NULL_FILL statement defines the character or hexadecimal value that PowerExchange uses to replace null characters in a field.

Data Sources: Nonrelational

Related Statements: NRDB_WRITE_NUM_NULL_FILL

Required: No

Syntax:

```
NRDB_WRITE_CHAR_NULL_FILL={character|_}
```

Value: For the *character* variable, enter any character or hexadecimal value. Default is the space () character.

Usage Notes: In the data map, define the field as a nullable field. Specify low values by using the hexadecimal format.

NRDB_WRITE_NUM_NULL_FILL Statement

The NRDB_WRITE_NUM_NULL_FILL statement defines the numeric or hexadecimal value that PowerExchange uses to replace null values in an unpacked numeric field.

Data Sources: Nonrelational

Related Statements: NRDB_WRITE_CHAR_NULL_FILL

Required: No

Syntax:

```
NRDB_WRITE_NUM_NULL_FILL={value|0}
```

Value: For the *value* variable, enter any numeric or a hexadecimal value. Default is 0.

Usage Notes: In the data map, define the unpacked numeric field as a nullable field. Specify low values by using the hexadecimal format.

NUMERICSIGN Statement

The NUMERICSIGN statement controls whether PowerExchange delivers unsigned fields as a positive X'C' or as unsigned X'F'.

Include the NUMERICSIGN statement in the DBMOVER configuration file on the client system.

Operating Systems: All

Required: No

Syntax:

```
NUMERICSIGN={C|F}
```

Valid Values:

- **C.** PowerExchange delivers unsigned fields as positive X'C'.
- **F.** For EBCDIC data, PowerExchange delivers unsigned fields as unsigned X'F'.

Default is C.

ODBASUPP Statement

The ODBASUPP statement controls whether PowerExchange can use the Open Database Access (ODBA) interface to access IMS databases and unload data sets for bulk data movement operations.

Operating Systems: z/OS

Data Sources: IMS

Related Statements: IMSID

Required: No

Syntax:

```
ODBASUPP={NO|YES}
```

Valid Values:

- **NO.** PowerExchange does not initialize the ODBA environment in the PowerExchange Listener address space. You must use DL/1 BATCH or BMP jobs to access IMS databases.
- **YES.** PowerExchange initializes the ODBA environment in the PowerExchange Listener address space. ODBA database queries run as PowerExchange Listener subtasks rather than as batch jobs.

Default is NO.

Usage Notes: To use ODBA to access IMS databases, you must complete the following tasks:

- Include the IMS RESLIB data set in either the STEPLIB DD concatenation of the PowerExchange Listener JCL or in the LNKST concatenation.
- Specify an IMS SSID in the IMSID statement in the DBMOVER member and in the data map. You can override this SSID for a database row test of a data map or when you run the PowerCenter bulk data movement session.

Specify a PSB name and PCB name in the data map. You can override these values for a database row test of a data map or when you run the PowerCenter bulk data movement session.

ORA_ERRORFILE Statement

The ORA_ERRORFILE statement specifies the name of the user-customized SQL error file that PowerExchange uses for Oracle bulk data movement operations.

Operating Systems: Linux, UNIX, and Windows

Data Sources: Oracle

Required: No

Syntax:

```
ORA_ERRORFILE=file_name
```

Value: For the *file_name* variable, enter the path and file name that contains the SQL error codes that you want PowerExchange to treat as either recoverable or fatal. PowerExchange provides a sample error action file called ora8err.act in the PowerExchange installation directory.

ORACLE_CAPTURE_TYPE Statement

The ORACLE_CAPTURE_TYPE statement specifies that the CDC type to use for the PowerExchange installation. The only available type is PowerExchange Express CDC for Oracle.

The ORACLE_CAPTURE_TYPE value must be consistent with the ORAD type of CAPI_CONNECTION statement that is defined on the system that initiates the connection to the Oracle system for change capture.

Operating Systems: Linux, UNIX, and Windows

Data Sources: Oracle

Required: No

Syntax:

```
ORACLE_CAPTURE_TYPE=D
```

Valid Values:

- **D.** PowerExchange uses Express CDC for Oracle and ORAD CAPI_CONNECTION statements.

No default value is available. The only valid value is D.

Usage Notes:

- PowerExchange Express CDC for Oracle handles all character columns as variable length columns. This behavioral difference affects PowerExchange column-level processing and the view of the CDC extraction map that is imported into PowerCenter. All systems that are involved in Oracle change capture processing must be aware of the Oracle CDC type that is in use

On the PowerExchange system where the ORAD CAPI_CONNECTION statements are defined, the CAPI_CONNECTION type implicitly defines the CDC type, and the ORACLE_CAPTURE_TYPE statement is optional. If other systems are involved in CDC processing, for example, because you run a separate PowerExchange Listener or use offload processing, you can define the ORACLE_CAPTURE_TYPE statement in the dbmover.cfg file on each system.

- For the ORAD CAPI_CONNECTION statement, the ORACLE_CAPTURE_TYPE value must be D. Otherwise, PowerExchange issues an error message and ends abnormally.

ORACLE_UNHANDLED_NUMASCHAR Statement

The ORACLE_UNHANDLED_NUMASCHAR statement controls how PowerExchange handles some numeric Oracle source columns.

If you enter Y, PowerExchange converts the following Oracle numeric datatypes:

- NUMBER columns that have a precision greater than 28 or an undefined length are treated as variable-length strings instead of double-precision floating-point numbers.
- FLOAT columns that have a precision greater than 15 significant digits are treated as variable-length strings.

PowerExchange uses the ORACLE_UNHANDLED_NUMASCHAR setting when creating capture registrations.

This statement applies to PowerExchange Express CDC for Oracle sources. You can use this statement to override PowerExchange default processing of numeric data to prevent data loss in certain circumstances. To override default processing, you must specify this statement prior to creating capture registrations.

Operating Systems: Linux, UNIX, and Windows

Data Sources: Oracle

Required: No

Syntax:

```
ORACLE_UNHANDLED_NUMASCHAR={Y|N}
```

Valid Values:

- **N.** PowerExchange uses its default processing of Oracle NUMBER data. If you have NUMBER columns that have a precision greater than 28 or an undefined length, or if you have FLOAT columns with a precision greater than 15, change data loss might occur.
- **Y.** PowerExchange handles NUMBER and FLOAT data in a manner that prevents data loss.

Default is N.

Usage Notes:

- Enter this parameter with a value of Y before you create capture registrations for the Oracle source tables that contain the NUMBER or FLOAT columns. If you enter this parameter after the registrations exist, you must set the status of the registrations to History and then create the registrations again. Otherwise, change data loss might occur.
- Oracle allows columns that have the NUMBER datatype to have their precision and scale determined by the numeric data that is written to the columns. Oracle supports a maximum precision of 38 and an exponent of +/-127.

If you do not explicitly define the precision and scale for NUMBER columns from which change data is captured, the following default PowerExchange and PowerCenter processing of change data can result in loss of precision and change data:

- PowerExchange handles data in NUMBER columns that have an undefined length or a length greater than 100 bytes as double-precision floating-point numbers.
- PowerCenter allows a maximum precision of 28 for decimal numbers.

To prevent change data loss with this type of data, enter Y for this statement and then create your capture registrations. PowerExchange registration processing can then handle numbers that have a precision greater than 28 as variable-length strings.

If you are writing the data to an Oracle target and want to maintain the precision as a variable-length string, edit the target definition to modify the column datatype. Within a PowerCenter mapping, you can

convert a variable-length string to a number either implicitly by connecting to a numeric port or explicitly by using expressions. To avoid loss of precision in implicit conversions, you might need to edit the mapping to pass the data as a string from source to target.

- PowerExchange supports the BINARY_DOUBLE and BINARY_FLOAT numeric datatypes by treating them as internal DOUBLE or FLOAT datatypes. However, PowerCenter converts BINARY_DOUBLE and BINARY_FLOAT datatypes to Oracle NUMBER(15) datatypes, which can result in arithmetic overflow and data loss.
- In columns with an Oracle numeric datatype, PowerExchange treats the value of infinity as 0.

ORACLECODEPAGE Statement

If the NLS_LANG environment variable specifies a character set other than UTF8 or AL32UTF8, you must define the ORACLECODEPAGE statement. This statement determines the code pages that PowerExchange and PowerCenter use for a specific Oracle database during bulk data movement operations.

The code page or pages that you specify in this statement must match the character set that is identified in the NLS_LANG environment variable. The NLS_LANG environment variable determines how the Oracle client libraries deliver data to PowerExchange.

Operating Systems: Linux, UNIX, and Windows

Data Sources: Oracle

Related Statements: CODEPAGE

Required: If NLS_LANG specifies a character set other than UTF8 or AL32UTF8

Syntax:

```
ORACLECODEPAGE=(tnsname_host
                  ,pwx_codepage
                  ,pc_codepage
                )
```

Parameters:

tnsname_host

Required. An entry in the Oracle tnsnames.ora configuration file for an Oracle database. The entry defines the database address that PowerExchange uses to connect to the database.

pwx_codepage

Required. A code page number or alias name that PowerExchange uses to identify a code page. To determine valid PowerExchange code page and alias values, use the ICUCHECK utility to generate report 5, "PowerExchange Code Page Names and Aliases."

Note: PowerExchange supports some common Oracle character set names as aliases to code pages.

pc_codepage

Optional. A name that controls the processing of the SQL statements that PowerExchange passes to Oracle on behalf of PowerCenter bulk data movement sessions. PowerExchange supplies a default that is usually suitable.

Do not specify this parameter except in special circumstances when the default does not work. For example, if you specify a user-defined ICU code page for the *pwx_codepage* parameter, you need to specify this parameter.

Usage Notes:

- You can specify up to 20 ORACLECODEPAGE statements, each for a separate database, in a dbmover.cfg configuration file.
- If PowerExchange uses the same NLS_LANG environment variable to access multiple Oracle databases, you do not need to specify a separate ORACLECODEPAGE statement for each database. Instead, specify a single ORACLECODEPAGE statement and leave the *tnsname_host* parameter blank. The specified code page then applies to all databases that have an entry in the tnsnames.ora file. The following example shows a statement without a *tnsname_host* parameter:

```
ORACLECODEPAGE=(,MS1252)
```

- If you enter an incorrect PowerCenter code page value, the ODLNumResultCols routine usually reports Oracle return code 911.

Example: If the NLS_LANG environment variable specifies Korean_Korea.KO16MSWIN949, define the following ORACLECODEPAGE statement:

```
ORACLECODEPAGE=(KO102DTL,MS949)
```

RELATED TOPICS:

- [“ICUCHECK Utility” on page 360](#)

ORACLEID Statement

The ORACLEID statement specifies the Oracle source instance, database, and connection information for CDC.

Operating Systems: Linux, UNIX, and Windows

Data Sources: Oracle CDC sources

Related Statements: CAPI_CONNECTION - ORAD

Required: Yes, for PowerExchange Express CDC for Oracle

Syntax:

```
ORACLEID=(collection_id
          ,oracle_db
          [,source_connect_string]
          [,capture_connect_string]
          [,fifth_positional_parameter]
          [,USEDATABASE])
```

Parameters:***collection_id***

Required. User-defined identifier for this ORACLEID statement. This value must match the ORACOLL parameter value in the ORAD CAPI_CONNECTION statement, the collection ID in the registration group defined for the source tables, and the DBID value in the PowerExchange Logger pwxocl configuration file.

Maximum length is eight characters.

oracle_db

Required. Name of the Oracle database that contains the source tables that are registered for change data capture. If you use PowerExchange Express CDC for Oracle to capture change data from a pluggable database (PDB) in an Oracle multitenant environment, this value is the name of the database that contains the PDB.

source_connect_string

Optional. Oracle connection string, defined in TNS, that is used to connect to the Oracle database that contains the source tables. This connection string must be defined in the Oracle Client tnsnames.ora file on the system with the source database.

For PowerExchange Express CDC for Oracle, the source connection string is used only for PowerExchange Navigator access to the Oracle source database. Enter this parameter in the dbmover configuration file on the machine from which the PowerExchange Listener retrieves data for PowerExchange Navigator requests. If you plan to run a database row test on extraction maps for the source tables, also specify the *capture_connect_string* parameter.

Note: The source connection string is not used to transfer change data.

If this value is null and the Oracle source is *not* a PDB in a multitenant environment, the value of the ORACLE_SID environment variable is used by default. If you use PowerExchange Express CDC for Oracle to capture change data from a PDBs, you must enter a value in this parameter.

capture_connect_string

Optional. Oracle connection string, defined in TNS, that the PowerExchange Logger uses to connect to the Oracle database with the source tables for PowerExchange Express CDCfor Oracle. This connection string must be specified in the Oracle Client tnsnames.ora file that is used for connection to the Oracle source database. If you use PowerExchange Express CDC to capture change data from a PDB in an Oracle multitenant environment, specify the name of the PDB service entry in the tnsnames.ora file.

If this value is null and the Oracle source is *not* a PDB in a multitenant environment, the value of the ORACLE_SID environment variable is used by default.

If this value is null and the Oracle source is a PDB, PowerExchange cannot capture change data for the source. If you use PowerExchange Express CDC for Oracle to capture change data from PDBs, you must enter a value in this parameter.

Also, for Express CDC for Oracle, if you have multiple Oracle databases and capture changes from a database other than the default database, you must specify both the *source_connect_string* and *capture_connect_string* parameters.

Tip: If possible, bypass the use of SQL*Net to improve PowerExchange Logger performance, even if the PowerExchange Logger is running on the same machine as the Oracle source database. Set the following environment variables, whenever possible, to enable connection to the appropriate Oracle database without using the *capture_connect_string* parameter and SQL*Net:

- ORACLE_HOME
- ORACLE_SID
- PATH
- On Linux or UNIX, one of the following:
 - LD_LIBRARY_PATH
 - LIBPATH
 - SHLIB_PATH

fifth_positional_parameter

Not used. Add a comma as a placeholder if you specify the USEDDBNAME positional parameter, for example:

```
ORACLEID=(collection_id,oracle_db,src_connect_string,capture_connect_string,,USEDDBNAME)
```

USEDATABASENAME

Optional. Specify this parameter only under all of the following conditions:

- You upgrade to PowerExchange 9.1.0 or later from an earlier release.
- You use Oracle 11g or later.
- You run the following SQL query on the V\$DATABASE view and the query returns different values for the NAME and DB_UNIQUE_DATABASE fields, including values that vary in case only such as ORAABC1 and oraabc1:

```
select name, db_unique_name from v$database;
```

In this situation, the USEDATABASENAME parameter can prevent potential restart errors that are caused by the difference in the NAME and DB_UNIQUE_DATABASE values.

Tip: Alternatively, you can specify the DB_UNIQUE_NAME value in the second positional parameter, *oracle_db*.

Usage Notes:

- PowerExchange requires an ORACLEID statement for each Oracle database for which you want to capture and extract change data. You can define a maximum of 20 ORACLEID statements in a single dbmover configuration file.
- Define the ORACLEID statement in the dbmover configuration file on the system where the PowerExchange Logger runs, or if you plan to perform Oracle CDC without the PowerExchange Logger, on the system where your PowerExchange extractions run.

OUSP Statement

The OUSP statement loads the z/OS OpenEdition security environment when using Adabas.

Use this statement at the direction of Informatica Global Customer Support. Also, use this statement if a netport job has an Adabas source that is encrypted with a cipher code.

Operating Systems: z/OS

Data Sources: Adabas

Related Statements: SECURITY

Required: No

Syntax:

```
OUSP={N | Y}
```

Valid Values:

- N. Default value.
- Y. For Adabas sources if SECURITY=(2,Y), set OUSP to Y at the direction of Informatica Global Customer Support. Also, set OUSP to Y if a netport job has an Adabas source that is encrypted with a cipher code.

Default is N.

PC_AUTH Statement

The PC_AUTH statement controls whether the PowerExchange Listener uses its MVS Program Call (PC) services routine to acquire the authorization to access DBMS load libraries.

Operating Systems: z/OS

Data Sources: All

Required: No

Syntax:

```
PC_AUTH={N|Y}
```

Valid Values:

- **N.** The PowerExchange Listener runs APF-authorized, and you must include APF-authorized copies of the DBMS load libraries in the STEPLIB DD statement of the PowerExchange Listener.
- **Y.** The PowerExchange Listener runs APF-authorized, and the PowerExchange Listener uses the PowerExchange-provided Program Call (PC) services routine to acquire the necessary authorization to access the DBMS load libraries.

Note: You must define the DBMS load libraries in the DTLLOAD DD statement instead of in the STEPLIB DD statement. If you use netport jobs to access data, define the load libraries in the DTLLOAD DD statement of the netport JCL.

Specify Y if you do not want to maintain any APF-authorized PowerExchange copies of the DBMS load libraries.

Default is N.

Usage Note:

- Use the PC_AUTH DBMOVER statement and the DTLLOAD DD statement to access load libraries that are not required to be APF-authorized. For example, use these statements to access CA IDMS/DB load libraries or to access a user-defined program that is invoked by the CallProg function in a data map to process source data in a record.

PIPE Statement

The PIPE statement defines the character that PowerExchange uses to separate fields in a TXT output file.

Data Sources: All

Required: No

Syntax:

```
PIPE={"character"|}|}
```

Value: For the *character* variable, enter a single character enclosed in double quotation marks. Default is the pipe (|) character.

POLLTIME Statement

The POLLTIME statement defines the amount of time, in thousandths of a second, that the PowerExchange Listener waits before polling for outstanding connections.

Operating Systems: All

Required: No

Syntax:

```
POLLTIME={wait_time|2000}
```

Value: For the *wait_time* variable, enter the amount of time, in thousandths of a second, that the PowerExchange Listener waits before polling for outstanding connections. The value 1000 equals one second. Valid values are from 250 through 15000. Default is 2000, or two seconds.

PRE861_COMPAT Statement

Controls whether PowerExchange automatically attempts to use offload processing.

Beginning in PowerExchange 8.6.1, PowerExchange automatically enables offload processing, provided that you use the PRE861_COMPAT=N statement that is supplied in the sample DBMOVER configuration file. For PowerExchange 8.6.1 or later, ensure that PRE861_COMPAT is set to N.

If you use a PowerExchange version earlier than 8.6.1, use PRE861_COMPAT=Y for backwards compatibility with earlier offload processing behavior.

Operating Systems: All

Data Sources: All

Required: Depends on the PowerExchange version

Syntax:

```
PRE861_COMPAT={Y|N}
```

Valid Values:

- **N.** Use this value for PowerExchange 8.6.1 and later. With this value, if you select **Auto** for the **Offload Processing** attribute on PWXPC connections, PowerExchange uses offload processing if possible. Also, PowerExchange ignores the **Offload Processing** option specified in VSAM and sequential data maps for previous releases.
- **Y.** Use this value only for PowerExchange versions earlier than 8.6.1. With this value, PowerExchange uses offload processing if you enable it. To enable offload processing for PWX DB2zOS relational connections or PWX CDC Real Time application connections, select **Yes** for the **Offload Processing** attribute. For PWX NRDB Batch application connections, select either **Filter Before** or **Filter After** for the **Offload Processing** attribute.

Note: If you use multiple types of PowerExchange data sources on different platforms, PowerCenter data previews and CDC sessions that use PWX connections might terminate with a connection error. To prevent this problem, define a CAPI_SRC_DEFAULT statement for each source type in the DBMOVER configuration file.

Default is Y.

PRGIND Statement

The PRGIND statement controls whether PowerExchange writes read progress messages to the PowerExchange message log file.

Data Sources: All

Related Statements: PRGINT

Required: No

Syntax:

```
PRGIND={N|Y}
```

Valid Values:

- **N.** PowerExchange does not write read progress messages to the PowerExchange message log file.
- **Y.** After reading the number of records specified in the PRGINT statement, PowerExchange writes PWX-04587 messages to the PowerExchange message log file. These messages indicate the number of records read by a CDC or bulk data movement session.

Default is N.

Usage Notes:

- For CDC sources and nonrelational bulk data movement sources, include the PRGIND and PRGINT statements in the DBMOVER configuration file on the PowerExchange Listener machine. For relational bulk data movement sources, include the PRGIND and PRGINT statements in the DBMOVER configuration file on the PowerCenter Integration Service machine.
- For DB2 relational connections, PowerExchange writes a PWX-04587 message after it reads the following number of records:
The smallest multiple of the **Array Size** connection attribute that is greater than or equal to the value of PRGINT

PRGINT Statement

The PRGINT statement defines the number of records that PowerExchange reads before writing read progress messages to the PowerExchange message log file.

Data Sources: All

Related Statements: PRGIND

Required: No

Syntax:

```
PRGINT={number_rows|250}
```

Value: Enter the number of records that PowerExchange must read before writing PWX-04587 messages to the PowerExchange message log file. For PowerExchange to write these message, you must also enter Y for the PRGIND statement.

Valid values are 0 through 2147483647. Default is 250.

Usage Notes:

- For CDC sources and nonrelational bulk data movement sources, include the PRGIND and PRGINT statements in the DBMOVER configuration file on the PowerExchange Listener machine. For relational bulk data movement sources, include the PRGIND and PRGINT statements in the DBMOVER configuration file on the PowerCenter Integration Service machine.
- For DB2 relational connections, PowerExchange writes a PWX-04587 message after it reads the following number of records:
The smallest multiple of the **Array Size** connection attribute that is greater than or equal to the value of PRGINT

PWXSOMAXCONN Statement

The PWXSOMAXCONN statement specifies the maximum number of TCP/IP socket connections that a PowerExchange Listener uses to listen for work.

Operating Systems: All

Required: No

Syntax:

```
PWXSOMAXCONN=num_connections
```

Value: For the *num_connections* variable, enter the maximum number of TCP/IP sockets connections that PowerExchange Listener uses to listen for work. Default is the SOMAXCONN value of the operating system. On z/OS, the default is 10.

Usage Notes: PowerExchange uses the lower of the two values PWXSOMAXCONN and SOMAXCONN. To increase the TCPIP queue depth, and thus the number of sockets available to PowerExchange tasks, you must increase both the TCPIP SOMAXCONN value and the PWXSOMAXCONN value.

RACF_CLASS Statement

The RACF_CLASS statement specifies the RACF class name that PowerExchange uses when checking PowerExchange-specific resource profiles.

Operating Systems: z/OS

Related Statements: DM_RESOURCE and SECURITY

Required: No

Syntax:

```
RACF_CLASS={class_name|FACILITY}
```

Value: For the *class_name* variable, enter the RACF class name. PowerExchange uses this class name when checking PowerExchange-specific resource profiles that control access to PowerExchange resources, such as capture registrations, data maps, and write access to certain data source types.

Default is the FACILITY class for all PowerExchange-specific resource profiles. If you specify a class other than FACILITY, define it in RACF or in your security product using a resource name that has a maximum length of 8 bytes.

RELATED TOPICS:

- [“z/OS Security” on page 263](#)

RDBMSINSRTDFLT Statement

The RDBMSINSRTDFLT statement controls whether PowerExchange uses default values for columns that you define with the WITH DEFAULT clause in an RDBMS.

Operating Systems: All

Data Sources: DB2, Microsoft SQL Server, MySQL, and Oracle targets

Required: No

Syntax:

```
RDBMSINSRTDFLT={N|Y}
```

Valid Values:

- **N.** PowerExchange uses PowerExchange defaults when writing data to columns that you define with the WITH DEFAULT clause.
- **Y.** PowerExchange uses RDBMS defaults when writing data to columns that you define with the WITH DEFAULT clause.

You must define the columns with a clause that enables the RDBMS to supply a default. Otherwise, an SQL error is generated.

Default is N.

RECFM Statement

The RECFM statement specifies the record format that PowerExchange uses when dynamically allocating target data sets on z/OS, if you do not provide RECFM information.

Operating Systems: z/OS

Related Statements: BS, DISP, LRECL, RELEASE, SPACE, UNIT, and VOLSER

Required: No

Syntax:

```
RECFM={E|FB|FBA|FU|V|VB|VBA|VU}
```

Valid Values:

- **E.** Fixed-length records.
- **FB.** Blocked fixed-length records.
- **FBA.** Blocked fixed-length records with ANSI control characters.
- **FU.** Undefined length records.
- **V.** Variable-length records.
- **VB.** Blocked variable-length records.
- **VBA.** Blocked variable-length records with ANSI control characters.
- **VU.** Undefined length records.

Default is F.

Usage Notes:

- If you specify either FU and VU for the RECFM value, PowerExchange uses RECFM=U in the dynamic allocation request.
- If you omit both the LRECL and RECFM statements, PowerExchange uses 32752 for the LRECL value and VB for the RECFM value when allocating data sets.

REJECT_FILE_DELIMITER Statement

The REJECT_FILE_DELIMITER statement defines the character that PowerExchange uses as the default delimiter character in a reject file.

This statement is useful if the data contains commas. For more information about reject files, see the *PowerExchange Bulk Data Movement Guide*.

Operating Systems: All

Required: No

Syntax:

```
REJECT_FILE_DELIMITER={character|_}
```

Value: For the *character* variable, enter a single character. Default is the comma (,) character.

Usage Notes: If the delimiter character must be the semicolon (;) character, enclose it in double quotation marks, as follows:

```
REJECT_FILE_DELIMITER=";"
```

RELEASE Statement

The RELEASE statement controls whether PowerExchange requests or does not request to release unused space for data sets that it dynamically allocates.

Operating Systems: z/OS

Related Statements: BS, DISP, LRECL, RECFM, SPACE, UNIT, and VOLSER

Required: No

Syntax:

```
RELEASE=Y
```

Value: Enter Y. For new data sets, PowerExchange requests the release of unused allocated space when the data set closes.

If you do not define this statement, PowerExchange does not request the release of unused space for data sets that it dynamically allocates.

RMTRDBDIRE Statement

The RMTRDBDIRE statement defines the name of the DB2 for i5/OS database on the local system that contains the source tables for CDC in a remote journaling environment.

Overrides the value specified in the optional RMTRDBDIRE parameter in the CRTPWXENV installation command. Include this statement if you plan to use remote journal receivers for CDC.

Operating Systems: i5/OS

Data Sources: DB2 for i5/OS CDC

Related Statements: RMTSYSNAME

Required: No

Syntax:

```
RMTRDBDIRE=database_name
```

Value: For the *database_name* variable, enter the name of the DB2 database on the local system that contains the DB2 source tables for CDC in a remote journaling environment. Define this value to i5/OS with the Work with Relational Database Directory Entries (WRKRDBDIRE) function. Maximum length is 18 characters.

Usage Notes: When you run the CRTPWXENV command during installation, the value in the RMTRDBDIRE statement is set to either the value specified in the optional RMTRDBDIRE parameter in the CRTPWXENV command or the default value of *LOCAL.

RMTSYSNAME Statement

The RMTSYSNAME statement defines the name of the i5/OS host system that contains the DB2 source tables for CDC, local journals, and journal receivers.

Also called the *local system*. Overrides the value specified in the optional RMTSYSNAME parameter in the CRTPWXENV installation command. Include this statement if you plan to use remote journal receivers for CDC.

Operating Systems: i5/OS

Data Sources: DB2 for i5/OS CDC

Related Statements: RMTRDBDIRE

Required: No

Syntax:

```
RMTSYSNAME=host_name
```

Value: For the *host_name* variable, enter the name of the i5/OS host system that contains the DB2 source tables for CDC, local journals, and journal receivers. Define this value to i5/OS with the Work with TCP/IP Host Table Entries function. To access the function, issue the CFGTCP (Configure TCP/IP) command. From the menu, select the option to work with the TCP/IP Host Table Entries function.

Maximum length is 68 characters.

Usage Notes: When you run the CRTPWENV command during installation, the value in the RMTSYSNAME statement is set to either the value specified in the optional RMTSYSNAME parameter in the CRTPWENV command or the default value of *NONE.

SECURITY Statement

The SECURITY statement controls PowerExchange user authentication and access to resources and commands.

Use the SECURITY statement in the DBMOVER configuration file to configure the following types of security:

- User authentication to access PowerExchange
- Access to files and data sets by PowerExchange jobs and tasks on z/OS and i5/OS
- User authorization to issue infacmd pwx commands to a PowerExchange application service in the Informatica domain
- User authorization to issue pwxcmd commands to a PowerExchange process
- User authorization to issue PowerExchange Listener LISTTASK and STOPTASK commands from the PowerExchange Navigator

Operating Systems: All

Related Statements: DM_RESOURCE, MVSDB2AF, and RACF_CLASS

Required: No

Syntax:

```
SECURITY=({Q|1|2}  
          ,{N|Y}  
          [,LDAP]  
          [, {ORACLE_LDAP|OPEN_LDAP}])
```

Parameters: The first positional parameter has the following valid values:

{Q|1|2}

Controls whether PowerExchange requires users to enter a valid operating system user ID and a password or passphrase. Also controls whether PowerExchange checks user-entered credentials to control access to file and database resources and the issuance of certain PowerExchange commands.

Enter one of the following options:

- **0.** PowerExchange does not require users to specify a valid operating system user ID and password and ignores any credentials that users supply.

On z/OS and i5/OS, PowerExchange uses the user ID under which the PowerExchange Listener or PowerExchange Condense task runs to control access to file resources. PowerExchange passes this user ID to the database system.

On Linux, UNIX, and Windows, PowerExchange uses the user ID under which the PowerExchange Listener task runs to control access to file resources. RDBMS security controls PowerExchange access to database resources based on the user ID that users specify on the PWX connection or in the PowerExchange Logger CAPTURE_NODE_UID parameter.

On all operating systems, PowerExchange does not check user authorization to issue commands. Any user can issue a command.

- **1.** On z/OS and i5/OS, PowerExchange requires users to specify a valid operating system user ID and a password or valid PowerExchange passphrase. PowerExchange checks these credentials when a PowerExchange task starts. Thereafter, PowerExchange controls access to file resources in the same manner as for option 0. For file access, PowerExchange uses the user ID under which the PowerExchange Listener or PowerExchange Condense task runs and passes this user ID to the database system.

On Linux, UNIX, and Windows, unless you specify LDAP for the third parameter of the SECURITY statement on supported systems, PowerExchange does not require users to specify a valid operating system user ID and password to access file or database resources and does not check for these credentials. As for option 0, PowerExchange uses the user ID under which the PowerExchange Listener task runs to control access to file resources. RDBMS security controls PowerExchange access to database resources based on the user ID that users specify on the PWX connection or in the PowerExchange Logger CAPTURE_NODE_UID parameter.

On all operating systems, PowerExchange does not check user authorization to issue commands. Any user can issue a command.

- **2.** Provides the most specific level of security.

- On z/OS, Informatica recommends that you use option 2. PowerExchange controls access based on 1) an MVS user ID and a password or valid PowerExchange passphrase and 2) the access control features of your z/OS security product, such as RACF or ACF2.

To read change data from the change stream, the ECCR must use a valid z/OS user ID and password or passphrase. The PowerExchange Listener checks these credentials when the ECCR task or job starts. To access the database to read data, PowerExchange passes the z/OS user ID and password or passphrase to the database system for database-specific security checking. In conjunction with the z/OS security product and MVS System Authorization Facility (SAF), PowerExchange checks the z/OS user ID and password or passphrase against the CAPX.REG.* resource profiles to control access to capture registrations.

To extract change data, run PowerCenter CDC sessions with a PWXPC connection that specifies a valid z/OS user ID and password or passphrase. For the session to access extraction maps, these user credentials must have READ access to the PowerExchange data set that is defined in the DTLCAMAP DD statement of the PowerExchange Listener JCL.

Note: A connection to DB2 for z/OS through the Call Attachment Facility (CAF) runs under the user ID of the PowerExchange Listener regardless of the security settings. DB2 uses the user ID that is specified on the connection only if the connection type is Recoverable Resource Manager Service Attachment Facility (RRSAF) or if offload processing is enabled.

PowerExchange also uses resource profiles to control who can run the following types of commands:

- pwxcmd commands for a PowerExchange Listener or PowerExchange Condense process that are issued from a Linux, UNIX, or Windows system
- PowerExchange Listener LISTTASK and STOPTASK commands that are issued from the PowerExchange Navigator or the DTLUTSK utility
- On i5/OS, PowerExchange requires users to specify a valid operating system user ID and password or passphrase. PowerExchange checks these credentials when a PowerExchange task starts. PowerExchange Listener subtask processes run under the supplied user ID and password or passphrase. PowerExchange uses this user ID and password or passphrase to control access to PowerExchange files. PowerExchange also passes this user ID and password or passphrase to the database system for data access.

PowerExchange uses security objects to control who can run the following types of commands:

- pwxcmd commands for a PowerExchange Listener or PowerExchange Condense process that are issued from a Linux, UNIX, or Windows system
- PowerExchange Listener LISTTASK and STOPTASK commands that are issued from the SNDLSTCMD interface, the PowerExchange Navigator, or the DTLUTSK utility
- On Linux, UNIX, and Windows, unless you specify LDAP for the third parameter of the SECURITY statement on supported systems, PowerExchange does not require users to specify an operating system ID and password to access PowerExchange files or a database. PowerExchange uses the user ID and password under which the PowerExchange Listener runs or that PowerExchange Logger for Linux, UNIX, and Windows uses to control access to PowerExchange files. RDBMS security controls access to the database.

However, you must specify a valid operating system user ID and password to run the following types of commands:

- An infacmd pwx command to a PowerExchange application service in the Informatica domain
- A pwxcmd command to a PowerExchange process

PowerExchange checks these user credentials against the USER and AUTHGROUP COMMANDS statements in the sign-on file to determine if a user is authorized to issue an infacmd pwx or pwxcmd command. In this case, the second positional parameter in the SECURITY statement is ignored.

Default is 0.

The second positional parameter has the following valid values:

{N|Y}

Controls use of PowerExchange selective sign-on file to authorize users to connect to the PowerExchange Listener.

Enter one of the following options:

- **N.** PowerExchange does not use the selective sign-on file.
- **Y.** PowerExchange uses the USER statement with the ALLOW and IP subparameters in the selective sign-on file to restrict users who can connect to the PowerExchange Listener.

Note: If you specify Y and also set the first parameter in the SECURITY statement to 1, PowerExchange uses the TASKCNTRL parameter in the USER statements in the sign-on file to control access to PowerExchange Listener LISTTASK and STOPTASK commands that are issued from the PowerExchange Navigator.

Default is N.

The optional third positional parameter has the following valid value:

LDAP

If you specify LDAP for the third positional parameter and specify 1 or 2 as the first positional parameter, PowerExchange uses LDAP authentication on supported Linux, UNIX, and Windows systems.

If you do not include the third parameter, PowerExchange does not use LDAP authentication.

The fourth positional parameter has the following valid values:

{ORACLE_LDAP|OPEN_LDAP}

If you specify LDAP for the third positional parameter, specifies which set of LDAP client libraries to load.

Enter one of the following options:

- **ORACLE_LDAP.** PowerExchange loads the Oracle LDAP client libraries.
Select this option only if you have an Oracle LDAP installation. PowerExchange does not provide the Oracle LDAP client libraries.
- **OPEN_LDAP.** PowerExchange loads the OpenLDAP client libraries.

Default is ORACLE_LDAP.

Usage Notes:

- In the z/OS Installation Assistant, if you click **Advanced Parms** on the **General Parameters** page, you can define the SECURITY_LEVEL and SECURITY_PWX parameters. The SECURITY_LEVEL parameter corresponds to the first parameter in the SECURITY statement. The SECURITY_PWX parameter corresponds to the second parameter in the SECURITY statement.
- On z/OS, when you set the first parameter of the SECURITY statement to 1 or 2, you must APF-authorize the STEPLIB for the PowerExchange Listener and netport jobs. Otherwise, PowerExchange cannot complete user authentication or control resource access, and instead operates as if you set this parameter to 0.
- If you offload column-level processing for a z/OS data source to the Linux, UNIX, or Windows system where the PowerCenter Integration Service runs, PowerCenter CDC sessions use the **Map Location User** and **Map Location Password** values that you specify on the connection to control access to all resources. The connection must be a PWX NRDB CDC application connection or PWX DB2zOS CDC application connection for which offload processing is enabled.
- If you log data from z/OS data sources to remote PowerExchange Logger for Linux, UNIX, and Windows log files, set the SECURITY option to 2 in the DBMOVER configuration member on z/OS. Ensure that the user ID and password in the PowerExchange Logger for Linux, UNIX, and windows configuration file, pwxccl, is a valid z/OS user ID and password that can pass z/OS security checking. To read captured data from the PowerExchange Logger for z/OS log files, these user credentials must have READ access to CAPX.REG.* resources profiles in the FACILITY class, which are managed by your z/OS security product. Also, for CDC sessions to extract data from the log files, the PWXPC connection must specify the z/OS user ID and password in the **Map Location User** and **Map Location Password** connection attributes. These user credential needs READ access to the CAPX.CND.* resource profiles.

RELATED TOPICS:

- [“i5/OS Security” on page 255](#)
- [“Linux, UNIX, and Windows Security” on page 260](#)
- [“z/OS Security” on page 263](#)
- [“PowerExchange Selective Sign-on” on page 293](#)

SECURITY_MSGSUPP Statement

The SECURITY_MSGSUPP statement controls whether to suppress the messages that are issued by RACF or a similar z/OS security product, such as CA Top Secret or ACF2, when verifying the logon credentials of a PowerExchange job or started task.

Note: For the PowerExchange Listener to use this parameter, you must also specify SECURITY=(2,x) or (1,x) in the same DBMOVER configuration file. If you specify SECURITY=(0,x), this parameter is ignored.

Operating Systems: z/OS

Related Statements: SECURITY

Required: No

Syntax:

```
SECURITY_MSGSUPP={Y|N}
```

Valid Values:

- **Y.** Suppress logon verification messages from RACF or a similar z/OS security product for PowerExchange jobs and started tasks. This option can help prevent flooding the system console with messages for successful logons.
- **N.** Allow RACF or a similar z/OS security product to write logon verification messages for PowerExchange jobs and started tasks. For example, RACF might issue the following ICH408I message for an invalid logon of a PowerExchange utility user to the operator console and JES Job Log:

```
13.55.14 STC00011 ICH408I USER(AUSER) GROUP(TEST) NAME(AUSER NAME1) 373
373 LOGON/JOB INITIATION - INVALID PASSWORD
13.55.14 STC00011 IRR013I VERIFICATION FAILED. INVALID PASSWORD GIVEN.
```

Default is Y.

SERVICE_TIMEOUT Statement

The SERVICE_TIMEOUT statement specifies the number of seconds that a PowerExchange Listener or Logger service waits to receive heartbeat data from the associated PowerExchange Listener or PowerExchange Logger before killing the service, shutting down, and issuing an error message.

Operating Systems: Linux, UNIX, and Windows

Required: No

Syntax:

```
SERVICE_TIMEOUT=(ISYNC,heartbeat_interval)
```

Parameters:

ISYNC

Required. Service to which the heartbeat interval applies. ISYNC is the only valid option.

heartbeat_interval

Number of seconds that a PowerExchange Listener or Logger service waits to receive heartbeat data from the associated PowerExchange Listener or PowerExchange Logger before killing the service, shutting down, and issuing an error message. If you have the PowerCenter high availability option, the domain then restarts the service.

Enter a number from 1 through 60. Default is 30.

SESSID Statement

The SESSID statement specifies the default value to use as the DB2 Correlation ID for DB2 requests.

Operating Systems: z/OS

Required: No

Syntax:

```
SESSID=correlation_id
```

Value: For *correlation_id*, enter a string of up to eight characters.

Usage Notes:

- To specify a default value for the DB2 Correlation ID for DB2 requests, include the SESSID statement in the DBMOVER configuration file on the PowerExchange Listener machine. To override this value for a specific DB2 for z/OS connection, specify the Correlation ID connection property in the PowerCenter Workflow Manager or the Informatica Developer tool.
- If no value is specified in either the client Correlation ID connection property or the SESSID statement in the DBMOVER configuration file on the PowerExchange Listener machine, PowerExchange uses the default value of DETAIL.
- The SESSID statement applies only if PowerExchange uses RRSAP to connect to DB2. To use RRSAP connections, include the MVSDB2AF statement in the DBMOVER configuration file.

SHOW_THREAD_PERF Statement

The SHOW_THREAD_PERF statement specifies the number of records that PowerExchange processes before writing statistics messages for sessions that use multithreaded processing or partitioned processing to the PowerExchange message log file or PowerCenter session log.

Operating Systems: All

Data Sources: All CDC sources. For bulk data movement sessions, the following sources: DB2 for z/OS tables, sequential and flat files, and VSAM data sets.

Data Targets: For bulk data movement sessions that use writer partitions, the following targets: VSAM data sets and sequential files.

Required: No

Syntax:

```
SHOW_THREAD_PERF=number_of_records
```

Value: For the *number_of_records* variable, enter the number of records that PowerExchange must process in a statistics reporting interval before writing messages with statistics that you can use to assess session performance. Valid values are 10000 through 50000000.

For sessions that use multithreading, PowerExchange writes messages PWX-31254 through PWX-31259 to the PowerExchange message log file. For bulk data movement sessions that use reader or writer partitioning, PWXPC writes messages PWX-31261 and PWX-31262 in the PowerCenter session log and PowerExchange message log.

Usage Notes: To have PWXPC write statistics messages in the PowerCenter session log, select the **Retrieve PWX log entries** attribute on a PWX CDC application connection or PWX Batch application connection. This attribute is selected by default.

SPACE Statement

The SPACE statement specifies the primary space and secondary space that PowerExchange uses to dynamically allocate target data sets on z/OS when you do not specify the SPACE parameter in the JCL.

Operating Systems: z/OS

Related Statements: BS, DISP, LRECL, RECFM, RELEASE, UNIT, and VOLSER

Required: No

Syntax:

```
SPACE=({unit_type|T}
        ,{primary_space|0}
        ,{secondary_space|0}
      )
```

Parameters:

space_unit

Required. The type of space units that PowerExchange uses when dynamically allocating target data sets on z/OS, if you do not specify space information.

Enter one of the following options:

- **C.** Cylinders.
- **T.** Tracks.

Default is T.

{primary_space|0}

Required. Primary space amount in the units that are defined in the *space_unit* parameter.

Valid values are from 1 through 16777215 tracks or the equivalent in cylinders. Default is 0. If you enter a value greater than 65535 tracks, coordinate with the space management team for the target system to verify that such a large allocation is supported.

{secondary_space|0}

Required. Secondary space amount in the units that are defined in the *space_unit* parameter.

Valid values are from 1 through 16777215 tracks or the equivalent in cylinders. Default is 0. If you enter a value greater than 65535 tracks, coordinate with the space management team for the target system to verify that such a large allocation is supported.

SSL Statement

The SSL statement specifies SSL certificate information for a Secure Sockets Layer (SSL) connection.

Operating Systems: Linux, UNIX, and Windows

Related Statements: SSL_ALLOW_SELFSIGNED, SSL_CIPHER_LIST, SSL_CONTEXT_METHOD, SSL_REQ_CLNT_CERT, and SSL_REQ_SRVR_CERT

Required: No

Syntax:

```
SSL=({PASS=client|EPASS=client_encrypted}
      ,KEY=key
      ,{CALIST=calist|CAPATH=directory}
    )
```

Parameters:

PASS=client

Required if you do not specify the EPASS option. The passphrase used to make an SSL connection.

EPASS=client_encrypted

Required if you do not specify the PASS option. The encrypted passphrase used to make a TLS connection.

Tip: You can create an encrypted passphrase in the PowerExchange Navigator by selecting **File > Encrypt Password**.

KEY=key

Required. The SSL key used to make an SSL connection. When this parameter is specified, the client keystore is validated during the initial connection handshake.

CALIST=calist

Required if you do not specify the CAPATH option. The Certificate Authority list (CALIST) to make an SSL connection. This is a file that contains the list of trusted certificates (truststore).

CAPATH=directory

Required if you do not specify the CALIST option. Specify the location of the truststore for trusted certificates:

- On Linux or UNIX, enter the trusted CA directory of the OpenSSL installation.
- On Windows, enter the certs directory of the OpenSSL installation.

RELATED TOPICS:

- [“Customizing the DBMOVER Configuration File on the SSL Server” on page 317](#)
- [“Customizing the DBMOVER Configuration File on the SSL Client” on page 318](#)

SSL_ALLOW_SELFSIGNED Statement

The SSL_ALLOW_SELFSIGNED statement specifies the type of certificate to use for SSL security.

You can use either self-signed certificates or certificates from a commercial certificate authority that the system trusts. Any certificate, whether it is self-signed or signed by a trusted CA, must be available in the CALIST file or CAPATH location.

Operating Systems: Linux, UNIX, and Windows

Related Statements: SSL, SSL_REQ_CLNT_CERT, and SSL_REQ_SRVR_CERT

Required: No

Syntax:

```
SSL_ALLOW_SELFSIGNED={N|Y}
```

Valid Values:

- **N.** Uses certificates from a commercial certificate authority (CA) that the system trusts.
- **Y.** Uses self-signed certificates. Self-signed certificates are not verifiable by following the chain to a trusted certificate authority. Self-signed certificates are for internal use only, such as connections within your organization's network or internal testing.

Default is N.

RELATED TOPICS:

- [“Customizing the DBMOVER Configuration File on the SSL Server” on page 317](#)
- [“Customizing the DBMOVER Configuration File on the SSL Client” on page 318](#)

SSL_CIPHER_LIST Statement

The `SSL_CIPHER_LIST` statement restricts the available cipher suites that a Linux, UNIX, or Windows client offers to a server during an SSL handshake to the specified list.

Operating Systems: Linux, UNIX, and Windows

Related Statements: `SSL`, `SSL_ALLOW_SELF_SIGNED`, `SSL_CONTEXT_METHOD`, `SSL_REQ_CLNT_CERT`, and `SSL_REQ_SRVR_CERT`

Required: No

Syntax:

```
SSL_CIPHER_LIST=cipher_list
```

Value: For the *cipher_list* variable, specify one or more OpenSSL cipher suite names, separated by commas.

The following table is a partial list of OpenSSL cipher suite names and the corresponding AT-TLS cipher suite names and hexadecimal values:

OpenSSL Cipher Suite Name	AT-TLS Cipher Suite Name	Hexadecimal Value
DHE-RSA-AES256-SHA	TLS_DHE_RSA_WITH_AES_256_CBC_SHA	39
DHE-DSS-AES256-SHA	TLS_DHE_DSS_WITH_AES_256_CBC_SHA	38
AES256-SHA	TLS_RSA_WITH_AES_256_CBC_SHA	35
EDH-RSA-DES-CBC3-SHA	TLS_DHE_RSA_WITH_3DES_EDE_CBC_SHA	16
EDH-DSS-DES-CBC3-SHA	TLS_DHE_DSS_WITH_3DES_EDE_CBC_SHA	13
DES-CBC3-SHA	TLS_RSA_WITH_3DES_EDE_CBC_SHA	0A
DHE-RSA-AES128-SHA	TLS_DHE_RSA_WITH_AES_128_CBC_SHA	33
DHE-DSS-AES128-SHA	TLS_DHE_DSS_WITH_AES_128_CBC_SHA	32
AES128-SHA	TLS_RSA_WITH_AES_128_CBC_SHA	2F

For a complete list of the cipher suites that are available in the OpenSSL cryptographic library on your Linux, UNIX, or Windows client machine, run the `REPORT_CIPHERS` command of the `PWXUSL` utility.

Usage Notes: You might include the `SSL_CIPHER_LIST` statement in the `DBMOVER` file on the client machine for any of the following reasons:

- To ensure that a Linux, UNIX, or Windows PowerExchange server never uses a weak cipher from a client machine.
- To force the use of a preferred cipher from a Linux, UNIX, or Windows client machine rather than having to change a `TTLSCipherParms` configuration statement on the z/OS server machine.
- To avoid the use of the Diffie-Hellman cipher on z/OS because of the slow connection time.

- To force the use of a weaker cipher, or a cipher with hardware assistance on z/OS, for faster performance.

SSL_CONTEXT_METHOD Statement

The SSL_CONTEXT_METHOD statement selects the TLS version that the peer supports for PowerExchange SSL/TLS communication.

Operating Systems: Linux, UNIX, and Windows

Related Statements: SSL, SSL_ALLOW_SELFSIGNED, SSL_CIPHER_LIST, SSL_REQ_CLNT_CERT, and SSL_REQ_SRVR_CERT

Required: No

Syntax:

```
SSL_CONTEXT_METHOD=context_method
```

Valid Values:

- **TLSV1.** The peer supports TLS version 1.
- **TLSV1_1.** The peer supports TLS version 1.1.
- **TLSV1_2.** The peer supports TLS version 1.2.

Default is TLSV1.

Usage Notes:

- Select the same values for the client and server.

SSL_REQ_CLNT_CERT Statement

The SSL_REQ_CLNT_CERT statement controls whether a PowerExchange server uses an SSL certificate to authenticate the identity of a PowerExchange client.

Operating Systems: Linux, UNIX, and Windows

Related Statements: SSL, SSL_ALLOW_SELFSIGNED, and SSL_REQ_SRVR_CERT

Required: No

Syntax:

```
SSL_REQ_CLNT_CERT={N|Y}
```

Valid Values:

- **N.** The PowerExchange server does not authenticate the identity of the PowerExchange client.
- **Y.** The PowerExchange server authenticates the identity of the PowerExchange client by using an SSL certificate.

Default is N for PowerExchange servers running on Linux, UNIX, or Windows. Default is Y for PowerExchange servers running on z/OS or IBM i.

RELATED TOPICS:

- [“Customizing the DBMOVER Configuration File on the SSL Server” on page 317](#)

SSL_REQ_SRVR_CERT Statement

The SSL_REQ_SRVR_CERT statement controls whether a PowerExchange client authenticates the identity of a PowerExchange server by using an SSL certificate.

Operating Systems: Linux, UNIX, and Windows

Related Statements: SSL, SSL_ALLOW_SELFSIGNED, and SSL_REQ_CLNT_CERT

Required: No

Syntax:

```
SSL_REQ_SRVR_CERT={N|Y}
```

Valid Values:

- **N.** The PowerExchange client does not authenticate the identity of the PowerExchange server.
- **Y.** The PowerExchange client authenticates the identity of the PowerExchange server by using an SSL certificate.

Default is N.

RELATED TOPICS:

- [“Customizing the DBMOVER Configuration File on the SSL Client” on page 318](#)

START_UP_USER_EXIT Statement

The START_UP_USER_EXIT statement specifies the name and programming language of a user-defined exit program that PowerExchange calls each time the PowerExchange Listener starts or shuts down.

Use this statement to enable the PowerExchange Listener to decrypt an Adabas database that is encrypted with an Adabas cipher code. The user exit program that you specify must provide a result set that includes the cipher code and some additional information.

Operating Systems: z/OS

Required: No

Syntax:

```
START_UP_USER_EXIT=(PROGRAM_NAME=program_name,LANGUAGE=language)
```

Parameters:

PROGRAM_NAME=*program_name*

Required. Name of the user exit program.

LANGUAGE=*language*

Required. Programming language in which the user exit program is written. Options are:

- **A.** Assembler language.
- **C.** C language.

Usage Notes:

- You can specify up to ten statements, each pointing to a different exit program. When the PowerExchange Listener starts, the user exit programs are executed in the order in which the associated START_UP_USER_EXIT statements occur in the DBMOVER configuration file. When the Listener shuts down, the user exit programs are executed in reverse order.

For more information about using the START_UP_USER_EXIT statement to decrypt an Adabas database that is encrypted with an Adabas cipher code, see the "Adabas Bulk Data Movement" chapter in the *PowerExchange Bulk Data Movement Guide*.

STATS Statement

The STATS statement controls whether PowerExchange writes SMF statistics records for the PowerExchange Listener to the z/OS System Management Facilities (SMF) or to a file. This statement also controls whether PowerExchange collects summary monitoring statistics on PowerExchange Listener resource usage and client requests and reports these statistics at a regular interval or only on demand.

SMF statistics are available for PowerExchange Listeners on z/OS. The PowerExchange Listener summary monitoring statistics are available for PowerExchange Listeners on i5/OS, Linux, UNIX, Windows, and z/OS.

Tip:

Operating Systems: i5/OS, Linux, UNIX, Windows, and z/OS

Required: No

Syntax:

```
STATS=(({SMF,record_number[, {interval|0}] |
        FILE,dataset_name[, {interval|0}] |
        MONITOR[, {interval|0}] |
        NONE})
)
```

Parameters:

SMF,record_number[, {interval|0}]

PowerExchange writes SMF statistics records to SMF.

Enter the following parameters:

record_number

Required. An SMF record number.

Valid values are 128 through 255.

{interval|0}

Optional. The interval, in minutes, after which PowerExchange writes an SMF record.

Valid values are 0 through 120. Default is 0, which disables interval processing. With the default, PowerExchange writes an SMF record at the end of a task or when the PowerExchange Listener shuts down.

FILE,dataset_name[, {interval|0}]

PowerExchange writes SMF statistics records to a sequential data set on z/OS.

Enter the following parameters:

dataset_name

Required. Name of the sequential data set to which PowerExchange writes SMF records. You must have pre-allocated the data set. Ensure that the data set is large enough to accommodate the SMF record sizes. For variable length, blocked data sets, the following DCB allocation attributes are usually sufficient: LRECL=5000 and BLKSIZE=27998.

{interval|0}

Optional. Interval, in minutes, after which PowerExchange writes an SMF record to the data set.

Valid values are 0 through 120. Default is 0, which disables interval processing. With the default, PowerExchange writes an SMF record when the PowerExchange Listener shuts down.

MONITOR[,*{interval|0}*]

PowerExchange collects summary statistics on PowerExchange Listener memory usage, CPU usage, client tasks, connections, data sent and received, and messages sent and received. These listener statistics are published to the system console or PowerExchange message log when you enter a DISPLAYSTATS or DISPLAYSTATS LISTENER command either locally from the command line or remotely with the pwxcmd program. If you configured a PowerExchange Listener Service in the Informatica domain, you can use the infacmd pwx displayStatsListener command to publish these statistics.

Note: For a PowerExchange Listener on i5/OS, you must use the pwxcmd displaystats command. You cannot enter the DISPLAYSTATS or DISPLAYSTATS LISTENER command from the i5/OS command line or use the infacmd pwx displayStatsListener command.

Optionally, for a PowerExchange Listener on any supported operating system, you can specify the following *interval* subparameter to publish the statistics at a regular interval:

{interval|0}

Optional. The interval, in minutes, after which PowerExchange publishes the PowerExchange Listener summary statistics. You can still use the DISPLAYSTATS, DISPLAYSTATS LISTENER, pwxcmd displaystats, or infacmd pwx displayStatsListener command to publish the statistics on demand.

Valid values are 0 through 120. Default is 0, which disables interval-based reporting of PowerExchange Listener summary statistics. With the default, PowerExchange writes these statistics only when a DISPLAYSTATS, pwxcmd displaystats, or infacmd pwx displayStatsListener command is issued.

NONE

PowerExchange does not write SMF records and does not collect PowerExchange Listener summary statistics.

Default is NONE.

Usage Notes:

- If you want to use the DISPLAYSTATS, DISPLAYSTATS LISTENER, or infacmd pwx displayStatsListener command to publish PowerExchange Listener summary statistics, you must include the MONITOR parameter in the STATS statement. For more information about the commands, see the *PowerExchange Command Reference* and *Informatica Command Reference*.
- To write records to SMF, you must APF-authorize all libraries in the STEPLIB concatenation of the PowerExchange Listener.
- Write PowerExchange SMF records to a file for test purposes or when you need to get some SMF statistics quickly. For long-term statistics collection, write PowerExchange SMF records to SMF.

- PowerExchange ships a data map, called pwxstat.file.dmp, with the Windows version of PowerExchange that you can use with PowerExchange Navigator to format the SMF records written to a file.

SUBMITTIMEOUT Statement

The SUBMITTIMEOUT statement specifies the time, in seconds, that a PowerExchange Listener waits to receive notification from a spawned batch job that it has started.

Operating Systems: z/OS

Related Statements: LOADJOBFILE and NETPORT

Required: No

Syntax:

```
SUBMITTIMEOUT={timeout_seconds|60}
```

Value: For the *timeout_seconds* variable, enter a number from 1 through 86400. Default is 60.

Usage Notes:

- By default, a PowerExchange Listener waits for 60 seconds for the spawned jobs to start. If this time expires, PowerExchange times out the job, stops the task in the PowerExchange Listener, and writes the PWX-00426 message to the PowerExchange message log.
- This statement applies to all batch jobs that a PowerExchange Listener spawns, which includes the following:
 - Netport jobs
 - DB2 LOAD utility jobs
 - CA IDMS/DB metadata jobs
 - Jobs submitted using the PROG=SUBMIT option of the DTLREXE utility

SUP_FUNC Statement

The SUP_FUNC statement defines the function name of the enclave SRB under which work is dispatched on a zIIP.

Include this statement only at the direction of Informatica Global Customer Support.

Operating Systems: z/OS

Related Statements: SUP_REQUEST, SUP_SSNAME, SUP_SSTYPE, USESUP, WORKCLASS

Data Sources: All

Required: No

Syntax:

```
SUP_FUNC=function_name
```

Value: Enter a name of up to eight characters.

Default is PWXFUNC.

For more information about offloading PowerExchange processing to a zIIP, see the following guides:

- *PowerExchange Bulk Data Movement Guide*
- *PowerExchange CDC Guide for z/OS*

SUP_REQUEST Statement

The SUP_REQUEST statement defines the name of the requestor for zIIP facilities.

Include this statement only at the direction of Informatica Global Customer Support.

Operating Systems: z/OS

Related Statements: SUP_FUNC, SUP_SSNAME, SUP_SSTYPE, USESUP, WORKCLASS

Data Sources: All

Required: No

Syntax:

```
SUP_REQUEST=requestor_name
```

Value: Enter a name of up to four characters.

Default is PWXR.

For more information about offloading PowerExchange processing to a zIIP, see the following guides:

- *PowerExchange Bulk Data Movement Guide*
- *PowerExchange CDC Guide for z/OS*

SUP_SSNAME Statement

The SUP_SSNAME statement defines the subsystem name that identifies the PowerExchange Listener started task to the IBM Workload Manager for offloading work to a zIIP. If your system includes multiple Listeners, you can define a different name for each Listener.

Operating Systems: z/OS

Related Statements: SUP_FUNC, SUP_REQUEST, SUP_SSTYPE, USESUP, WORKCLASS

Data Sources: All

Required: No

Syntax:

```
SUP_SSNAME=subsystem_name
```

Value: Enter a name of up to eight characters.

Default is PWXLSTNR.

For more information about offloading PowerExchange processing to a zIIP, see the following guides:

- *PowerExchange Bulk Data Movement Guide*
- *PowerExchange CDC Guide for z/OS*

SUP_SSTYPE Statement

The SUP_SSTYPE statement defines the name that the IBM Workload Manager uses as the subsystem type for the enclave SRB under which work is dispatched on the zIIP.

Operating Systems: z/OS

Related Statements: SUP_FUNC, SUP_REQUEST, SUP_SSNAME, USESUP, WORKCLASS

Data Sources: All

Required: No

Syntax:

```
SUP_SSTYPE=subsystem_type
```

Value: Enter a name of up to four characters.

Default is PWX.

For more information about offloading PowerExchange processing to a zIIP, see the following guides:

- *PowerExchange Bulk Data Movement Guide*
- *PowerExchange CDC Guide for z/OS*

SUPPRESS_DATA_LOGGING Statement

The SUPPRESS_DATA_LOGGING statement controls whether PowerExchange writes a message that displays the first 40 characters of any row that fails the data checking options specified in a data map.

Operating Systems: All

Required: No

Syntax:

```
SUPPRESS_DATA_LOGGING=Y
```

Value: Enter Y. PowerExchange writes the PWX-03006 message to the PowerExchange message log file for any row that fails the data checking options specified in the data map. This message indicates that PowerExchange suppressed message PWX-02152, which displays the first 40 characters of the row that failed data checking.

Usage Notes: By default, PowerExchange writes the PWX-02152 message to the PowerExchange message log for any row that fails the data checking options specified in the data map. This message displays the first 40 characters of the row that failed data checking.

To suppress the logging of data in this manner, you can select the **Suppress Data Logging** option on the **Data Checking** tab in the **Data Map Properties** dialog box.

SVCNODE Statement

The SVCNODE statement specifies the TCP/IP port on which a PowerExchange process listens for commands. Process types include the PowerExchange Listener, PowerExchange Condense, PowerExchange log-based or table-based ECCRs, and the PowerExchange Logger for Linux, Unix and Windows.

You can issue commands to a PowerExchange process through the following programs:

- Use the infacmd pwx program to issue a command to a process on Linux, UNIX, or Windows.
- Use the pwxcmd program to issue a command to a process on i5/OS, Linux, UNIX, Windows, or z/OS.

Operating Systems: All

Related Statements: CMDNODE, LISTENER, and NODE

Required: No

Syntax:

```
SVCNODE=(service_name  
         ,listen_port  
)
```

Parameters:***service_name***

Required. PowerExchange service name of up to 12 characters. This name must match one of the following values:

- For a PowerExchange Condense process, use the service name that you specified in the CONDENSENAME statement in the CAPTPARM configuration member or file.
- For a PowerExchange Adabas, IDMS, or IMS log-based ECCR process or a Datacom table-based ECCR process on z/OS, use the ECCR name that you specified in the CMDNODE statement in the DBMOVER configuration file.
- For a PowerExchange Logger for Linux, UNIX, and Windows process, use the service name that you specified in the CONDENSENAME statement in the pwxcl.cfg file.
- For a PowerExchange Listener process, use the name that you specified for the *listener_node* parameter in the LISTENER statement in the DBMOVER configuration file.

To issue infacmd pwx commands to connect to the Listener through the Listener application service, this name must match one of the following values:

- If you created the application service through Informatica Administrator, use the service name that you specified in the **Start Parameters** property.
- If you created the application service through the infacmd pwx CreateListenerService command, use the name that you specified for the *service_name* parameter of the -StartParameters option on the command.

To issue infacmd pwx commands to connect to the PowerExchange Logger for Linux, UNIX, and Windows through the Logger application service, ensure that the pwxcl.cfg file that the Logger Service uses includes the correct value for CONDENSENAME. When you create the Logger Service, you can optionally specify a configuration file different from the default pwxcl.cfg file in the installation directory, as follows:

- If you created the application service through Informatica Administrator, specify the cs value in the **Start Parameters** property.
- If you created the application service through the infacmd pwx CreateLoggerService command, specify the cs value in the -StartParameters option on the command.

listen_port

Required. Unique port number on which the PowerExchange Listener, PowerExchange log-based or table-based ECCR, PowerExchange Logger for Linux, UNIX, and Windows, or PowerExchange Condense listens for commands.

To issue infacmd pwx commands to connect to the Listener or Logger through the Listener or Logger application service, the port number that you specify in the SVCNODE statement must match one of the following values:

- If you created the application service through Informatica Administrator, the value that you specified for the **SVCNODE Port Number** property.
- If you created the application service through the infacmd pwx CreateListenerService or CreateLoggerService command, the value that you specified for the -SvcPort option on the command.

SYSOUT_TIMESTAMP Statement

The SYSOUT_TIMESTAMP statement controls whether PowerExchange includes time stamp information in output files and console output.

By default, PowerExchange includes time stamp information on every message written to the PowerExchange message log file. This statement does not affect messages written to the PowerExchange message log file.

Operating Systems: All

Required: No

Syntax:

```
SYSOUT_TIMESTAMP={N|Y}
```

Valid Values:

- **N.** PowerExchange does not include time stamp information at the beginning of messages written to SYSOUT files on z/OS, QPRINT files on i5/OS, and to the console on Linux, UNIX, and Windows.
- **Y.** PowerExchange does include time stamp information at the beginning of messages written to SYSOUT files on z/OS, QPRINT files on i5/OS, and to the console on Linux, UNIX, and Windows.

The time stamp information has the following format:

```
ddmmyy hhmmss
```

Default is N.

TAPEWAIT Statement

The TAPEWAIT statement controls whether netport jobs wait or do not wait for tape volumes.

Operating Systems: z/OS

Related Statements: GDGLOCATE, NETPORT, and WAITDSN

Required: No

Syntax:

```
TAPEWAIT=Y
```

Value: Enter Y. PowerExchange waits for tape volumes by specifying the S99WTVOL bit in the S99FLAG2 indicators on dynamic allocation (SVC99) requests.

Usage Notes:

- By default, PowerExchange does not wait for volumes.
- To use this statement, you must APF-authorize the libraries in the STEPLIB of the netport JCL.
- PowerExchange ignores this statement when initializing the PowerExchange Listener.

TCPIP_ACCEPT_NONBLOCK Statement

The TCPIP_ACCEPT_NONBLOCK statement specifies whether the TCP/IP ports on which the PowerExchange Listener listens are blocking or nonblocking ports.

Define TCPIP_ACCEPT_NONBLOCK=Y if the following conditions apply:

- You perform a security check by using a port scanner such as Nessus.
- The PowerExchange Listener appears to hang indefinitely on the main Listener port or ignore commands.
- A Listener port for a netport job is defined.

Operating Systems: z/OS

Required: No

Syntax:

```
TCPIP_ACCEPT_NONBLOCK={N|Y}
```

Valid Values:

- **N.** PowerExchange Listener listening ports are blocking ports. When a blocking port is accessed, it blocks other attempts to access the port.
- **Y.** PowerExchange Listener listening ports are nonblocking ports.

Default is N.

Usage Notes: If you perform a security check by using a port scanner such as Nessus, the PowerExchange Listener might hang indefinitely on the main Listener port or ignore commands if a Listener port for a netport job is also defined. The problem occurs when a call to the TCPIP accept() function on the netport port is blocked until a connection is present. While the netport port is blocked, the main PowerExchange Listener port is never polled, so that non-netport requests appear to be hung. The Listener remains in this state until a subsequent netport request unblocks the netport port.

If this situation occurs, include the TCPIP_ACCEPT_NONBLOCK=Y statement in the DBMOVER configuration file on the Listener machine. This statement makes the PowerExchange Listener listening ports nonblocking, so that the accept() call can return a suitable status if it fails.

TCPIP_DIAGNOSTICS_TRACE Statement

The TCPIP_DIAGNOSTICS_TRACE statement controls the issuing of network diagnostic messages PWX-33316, PWX-33317, and PWX-33318. These messages provide information about network send and receive operations that occur before an error condition or the termination of a connection.

Operating Systems: All

Related Statements: TCPIP_DIAGNOSTICS_TRACE_SZ

Required: No

Syntax:

```
TCPIP_DIAGNOSTICS_TRACE={A|E|N}
```

Valid Values:

- **A.** PowerExchange always issues network diagnostic messages when the connection ends.
- **E.** PowerExchange issues network diagnostic messages when a network-related error occurs.
- **N.** PowerExchange does not issue network diagnostic messages.

Default is E.

TCPIP_DIAGNOSTICS_TRACE_SZ Statement

The TCPIP_DIAGNOSTICS_TRACE_SZ statement controls the number of trace records to keep for network send and receive operations. PowerExchange reports the information in these records in messages PWX-33316 and PWX-33317. These messages provide information about network send and receive operations that occur before an error condition or the termination of a connection.

Operating Systems: All

Related Statements: TCPIP_DIAGNOSTICS_TRACE

Required: No

Syntax:

```
TCPIP_DIAGNOSTICS_TRACE_SZ={size|20}
```

Value: Enter a value from 5 through 99999. Default is 20.

TCPIP_SHOW_POOLING Statement

The TCPIP_SHOW_POOLING statement specifies whether to write diagnostic information about connection pooling to the PowerExchange log file.

If this option is enabled in the client-side DBMOVER file, PowerExchange issues message PWX-33805.

Operating Systems: All

Required: No

Syntax:

```
TCPIP_SHOW_POOLING={N|Y}
```

Valid Values:

- **N.** PowerExchange does not issue message PWX-33805.
- **Y.** PowerExchange issues message PWX-33805.

Default is N.

Usage Notes: Message PWX-33805 provides the following information:

- **Size.** Size of connection pool.
- **Hits.** Number of times that PowerExchange found a connection in the PowerExchange connection pool that it could reuse.
- **Partial hits.** Number of times that PowerExchange found a connection in the PowerExchange connection pool that it could modify and reuse.
- **Misses.** Number of times that PowerExchange could not find a connection in the PowerExchange connection pool that it could reuse.
- **Expired.** Number of connections that were discarded from the PowerExchange connection pool because the maximum idle time was exceeded.
- **Discarded pool full.** Number of connections that were discarded from the PowerExchange connection pool because the pool was full.
- **Discarded error.** Number of connections that were discarded from the PowerExchange connection pool due to an error condition.

TCPIPBUFSIZE Statement

The TCPIPBUFSIZE statement is deprecated and exists for backward compatibility only.

TCPIPVER Statement

The TCPIPVER statement specifies alternative TCP/IP socket code that PowerExchange uses instead of the standard IBM TCP/IP UNIX socket code.

By default, PowerExchange uses IBM TCP/IP UNIX sockets. Define this statement if you use CA TCPAccess.

Operating Systems: z/OS

Required: No

Syntax:

```
TCPIPVER={2|3}
```

Valid Values:

- **2.** PowerExchange uses CA TCPAccess socket code on z/OS instead of IBM TCP/IP UNIX socket code.
- **3.** PowerExchange uses native MVS socket code. Specify this value at the direction of Informatica Global Customer Support.

TEMPHLQ Statement

The TEMPHLQ statement overrides the high-level qualifier that PowerExchange uses by default when creating a temporary file for CA IDMS/DB metadata.

If you do not want PowerExchange to create data sets with the PowerExchange Listener user ID, define this statement. PowerExchange ignores the TEMPHLQ statement when you specify 2 for the first parameter of the SECURITY statement.

Operating Systems: z/OS

Data Sources: CA IDMS/DB

Related Statements: SECURITY

Required: No

Syntax:

```
TEMPHLQ=hlq
```

Value: For the *hlq* variable, enter the high-level qualifier (HLQ) for temporary data sets that PowerExchange creates for CA IDMS/DB metadata. A valid value is a 1- to 17-character string. By default, PowerExchange uses the PowerExchange Listener user ID as the HLQ for the metadata temporary data sets.

To use the TEMPHLQ statement, you must also specify 0 or 1 in the first parameter in the SECURITY statement.

Example: If you define the following SECURITY and TEMPHLQ statements:

```
SECURITY=(0,N)  
TEMPHLQ=B.C
```

PowerExchange creates the following data set during the IDMS copybook import process to hold the CA IDMS/DB metadata:

```
B.C.METADATA.DTL000001
```

TEXT_EOF_FOR_BINARY Statement

The TEXT_EOF_FOR_BINARY statement controls whether PowerExchange ignores hexadecimal value 1A when reading binary flat files on Linux, UNIX, or Windows.

Operating Systems: Linux, UNIX, and Windows

Data Sources: Binary flat files

Required: No

Syntax:

```
TEXT_EOF_FOR_BINARY={N|Y}
```

Valid Values:

- **N.** PowerExchange ignores hexadecimal value 1A when reading binary flat files on Linux, UNIX, and Windows.
- **Y.** PowerExchange processes hexadecimal value 1A as an end-of-file (EOF) marker when reading binary flat files on Linux, UNIX, and Windows.

Default is N.

TIMEZONE Statement

The TIMEZONE statement enables PowerExchange to read or write Adabas date-time values in the user's local time for bulk data movement sessions. It specifies the local time zone that PowerExchange uses.

This statement applies only to Adabas 8.2.2 data sources or targets that include date-time fields for which the Adabas TZ option is specified in the field definitions.

If you need to maintain consistent date-time values across CDC and bulk data movement sessions that process the same Adabas sources or targets, do not specify this statement.

Operating Systems: z/OS

Data Sources or Targets: Adabas

Required: No

Syntax:

```
TIMEZONE=area/location
```

Value: For the *area/location* string, enter the local time zone value from the tz database. This value is case sensitive.

Usage Notes:

- For PowerExchange to read or write date-time values in the user's local time, the user's local time zone must be defined in the TZINFO member of the Adabas source library.
- If you do not use the TIMEZONE statement, PowerExchange reads and writes Adabas date-time values in Coordinated Universal Time (UTC). UTC is the time in which Adabas stores date-time values when the TZ option is specified.
- For CDC sessions, the TIMEZONE statement is ignored and PowerExchange always reads and writes Adabas date-time values in UTC.
- **IMPORTANT:** PowerExchange cannot convert UTC date-time values to the local time for CDC sessions. If you run a CDC session and a bulk data movement session for which you specified a local time zone against the same Adabas sources or targets, these sessions might have inconsistent date-time values.

TRACE Statement

The TRACE statement activates traces in PowerExchange code paths for diagnostic purposes.

Operating Systems: All

Required: No

The TRACE statement can severely impact performance. Use this statement at the direction of Informatica Global Customer Support.

TRACING Statement

The TRACING statement enables PowerExchange alternative logging and specifies attributes for the alternative log files. PowerExchange uses the alternative log files instead of the default PowerExchange message log file to store messages.

Operating Systems: All

Related Statements: LOGPATH

Required: No

Syntax:

```
TRACING=(PFX=prefix
        [,APPEND={N|Y}]
        [,BUFFERS={number_of_buffers|100}]
        [,FILENUM={number_of_files|5}]
        [,FLUSH={flush_interval|99}]
        [,RECLEN={record_length|80}]
        [,SIZE={log_size|100}]
        [,VIEW={N|Y}]
)
```

Parameters:

PFX=*prefix*

Required. Specifies the prefix for the alternative log file names.

PowerExchange uses the following system-based rules to create the alternative log file names:

i5/OS

PowerExchange uses the PFX value to create the member names of the log files in the PowerExchange data library. The generated log file names vary based on whether the PowerExchange Listener, PowerExchange Condense, or other PowerExchange jobs create the files.

- The PowerExchange Listener uses the following file naming convention:

datalib/Plistener_port(prefixnnn)

- PowerExchange Condense and other PowerExchange jobs use the following file naming convention:

datalib/JOBjob_number(prefixnnn)

These naming conventions include the following variables:

- *datalib* is the PowerExchange data library name specified during PowerExchange installation.
- *listener_port* is the PowerExchange Listener port number.
- *job_number* is the i5/OS job number for the tracing subtask, DTLTRTSK, that runs under PowerExchange Condense or other PowerExchange jobs.

- *prefixnnn* is the PFX parameter value with an appended sequential number from 001 through 999.

For example, a PowerExchange Listener that has a listener port number of 2480, a PFX value of PWXLOG, and a FILENUM value of 3 creates the following log files:

```
datalib/P02480 (PWXLOG001)
datalib/P02480 (PWXLOG002)
datalib/P02480 (PWXLOG003)
```

Maximum length for the PFX value is seven characters.

Linux, UNIX, and Windows

PowerExchange uses the PFX value as the subdirectory name in which to place the log files.

PowerExchange uses the LOGPATH statement in the dbmover.cfg file to determine the directory in which to place this log subdirectory.

The generated log file names vary based on whether the PowerExchange Listener, PowerExchange Condense, or other PowerExchange tasks create the files.

- The PowerExchange Listener uses the following file-naming convention:

Linux and UNIX:

```
logpath/prefix/DTLLST1.plistener_port.nnnn.log
```

Windows Listener Service:

```
logpath\prefix\DTLLSTNT.plistener_port.nnnn.log
```

Windows Listener:

```
logpath\prefix\DTLLST1.plistener_port.nnnn.log
```

- The PowerExchange Logger for Linux, UNIX, and Windows uses the following file-naming convention:

Linux and UNIX:

```
logpath/prefix/PWXCLL.tyyyymmddhhmmss.ppid.nnnn.log
```

Windows:

```
logpath\prefix\PWXCLL.tyyyymmddhhmmss.ppid.nnnn.log
```

- For other tasks, PowerExchange uses the following file-naming convention:

Linux and UNIX:

```
logpath/prefix/module.tyyyymmddhhmmss.ppid.nnnn.log
```

Windows:

```
logpath\prefix\module.tyyyymmddhhmmss.ppid.nnnn.log
```

The variables represent the following values:

- *logpath* is the value of the LOGPATH statement in the dbmover.cfg file.
- *prefix* is the PFX parameter value.
- *module* is the name of the PowerExchange module that is running, such as DTLURDMO for that utility or DTLOBCDRVR for PowerCenter operations.
- *listener_port* is the PowerExchange Listener port number.
- *yyyymmddhhmmss* is the time stamp when the file was created.
- *pid* is the process ID of the PowerExchange task.
- *nnn* is a sequential number from 001 through 999.

For example, a PowerExchange Listener that runs on UNIX with a port number of 2480, a PFX value of PWXLOG, and a FILENUM value of 3 creates the following log files:

```
logpath/PWXLOG/DTLLST1.p02480.n001.log  
logpath/PWXLOG/DTLLST1.p02480.n002.log  
logpath/PWXLOG/DTLLST1.p02480.n003.log
```

Maximum length for the PFX value is 210 characters.

z/OS

PowerExchange uses the PFX value as the high-level qualifier or qualifiers (HLQ) for dynamically allocated alternative log data sets. These data sets are sequential data sets. Alternatively, you can specify DTLLOGnn DD statements in the JCL for a PowerExchange task to allocate the alternative log data sets. By default, PowerExchange uses dynamically allocated alternative log data sets.

If you use a DTLLOGnn DD statement that allocates an extended sequential data set, PowerExchange writes only one message on each track. If the DD statement allocates a normal sequential data set, PowerExchange writes one message to each data block.

Note: Do not use DFSMS compression for alternative log data sets.

For dynamically allocated log data sets, the generated data set names vary based on whether the PowerExchange Listener or other PowerExchange batch jobs or started tasks create the files.

- The PowerExchange Listener uses the following file naming convention:

```
prefix.sysid.Plistener_port.Nnnn
```

- All other PowerExchange batch jobs and started tasks use the following file naming convention:

```
prefix.job_name.job_number.sysid.Nnnn
```

The variables represent the following values:

- *prefix* is the high-level qualifier or qualifiers that you specify in the PFX parameter. Maximum length of the entire prefix is 16 characters.
- *sysid* is the system ID of the z/OS system on which the batch job or started task runs.
- *listener_port* is the PowerExchange Listener port number.
- *job_name* is the job name of the batch job or started task.
- *job_number* is the JES job number, which begins with JOB for batch jobs and STC for started tasks.
- *nnn* is a generated sequential number from 001 through 999.

For example, a PowerExchange Listener that runs on the MVS1 system with a port number of 2480, a PFX value of PWXLOG, and a FILENUM value of 3 creates the following log files:

```
PWXLOG.MVS1.P02480.N001  
PWXLOG.MVS1.P02480.N002  
PWXLOG.MVS1.P02480.N003
```

APPEND={N|Y}

Optional. Controls how PowerExchange uses the message log files when the PowerExchange component that issues messages is restarted.

Options are:

- **N.** PowerExchange opens a new log file or overwrites the oldest log file as the log file.

For example, if you set FILENUM=3 to use three dynamically allocated log files, when the PowerExchange Listener starts, it tries to open log file 1, 2, and 3, in that order. Then, PowerExchange completes one of the following actions:

- If one or more of the dynamically allocated log files do not exist, PowerExchange uses the first nonexistent log file as the initial log file. For example, if log files 1 and 2 exist but log file 3 does not exist, PowerExchange uses log file 3 as the initial log file. If no log files exist, PowerExchange uses log file 1 as the initial log file.
- If all three log files exist, PowerExchange uses the oldest log file as the initial log file, completely overwriting it.

Note: If you use a GDG on z/OS for alternative logging, PowerExchange creates a new generation when the PowerExchange Listener starts.

- **Y.** PowerExchange opens the most recent log file, if one exists, and appends log messages to the end of it. If no log files exist, PowerExchange opens a new log file.

For example, if you set FILENUM=3 to use three log files, when the PowerExchange Listener starts, it tries to open log file 1, 2, and 3, in that order. Then, PowerExchange completes one of the following actions:

- If one or more log files exist, PowerExchange opens the most recent log file and appends log messages to the end of it.

If you use a GDG for alternative logging on z/OS and specify GDG(0) in the DTLLOGnn DD statement of the PowerExchange Listener JCL, PowerExchange appends messages to the end of the current generation. If you do not use GDG(0), PowerExchange ignores this parameter.

- If no log files exist, PowerExchange opens a new log file 1 and uses it as the log file.

Default is Y.

BUFFERS={number_of_buffers|100}

Optional. Specifies the number of buffers that PowerExchange allocates to receive message and trace information from PowerExchange subtasks. If the buffer space is full, the PowerExchange subtasks that generate message and trace information wait until buffer space is available. PowerExchange programs use this buffer space internally.

Valid values are 5 through 9999. Default is 100.

Specify this parameter only at the direction of Informatica Global Customer Support.

FILENUM={number_of_files|5}

Optional. Specifies the number of alternative log files that PowerExchange creates and uses when the log files are dynamically allocated. When a log file becomes full, PowerExchange switches to the oldest alternative log file and overwrites it.

Valid values are from 1 through 99. Default is 5.

Note: On z/OS, the FILENUM parameter is ignored if you use a GDG for alternative logging or if you specify DTLLOGnn DD statements in the JCL for a PowerExchange component that issues DTLLOG messages. For a GDG, the parameter is ignored regardless of whether you specify GDG(0) or GDG(+1) in the DTLLOGnn DD statement in the JCL.

FLUSH={flush_interval|99}

Optional. Specifies the number of log records that PowerExchange collects before it flushes them to the log file on disk. PowerExchange must periodically flush log records to enable PowerExchange to recover from out-of-space conditions. Low flush values result in more I/O activity to the log file.

Valid values are 1 through 99. Default is 99.

RECLEN={record_length|80}

Optional. Specifies the record length that PowerExchange uses to write log records to the log file. PowerExchange writes the log record on multiple lines if the length of the message exceeds the record length.

Valid values are 80 through 255. Default is 80.

Note: If you do not specify the RECLEN parameter and if you enter a value greater than 80 in the LOG_LINE_LIMIT statement, PowerExchange uses the LOG_LINE_LIMIT value as the RECLEN value.

SIZE={log_size|100}

Optional. Specifies the approximate amount of log data, in megabytes, that PowerExchange writes to an alternative log file. After PowerExchange reaches this value, it closes the current log file and opens the next log file to continue writing log records.

Valid values are 1 through 2048. Default is 100.

Note: On z/OS, if a manually allocated data set is larger than the SIZE value, PowerExchange limits the amount of log data that it writes to the data set to the SIZE value. If the data set is smaller than the SIZE value, the data set size limits the amount of log data that PowerExchange can write to it. When an out-of-space condition occurs, PowerExchange switches to the next manually allocated message data set.

VIEW={N|Y}

Optional. Controls whether PowerExchange periodically closes and reopens the current log file when the FLUSH interval expires. You can specify this parameter on all operating systems, but it is most useful on z/OS. On z/OS, you cannot view the alternative message log records until the log data set is closed. On operating systems other than z/OS, you can view the log records after PowerExchange flushes the log records to disk based on the FLUSH interval. The current log file does not need to be closed for you to view the log records.

Options are:

- **N.** PowerExchange does not periodically close and reopen the current log file.
- **Y.** PowerExchange periodically closes and reopens the current log file.

Tip: On z/OS, Informatica recommends that you specify VIEW=Y to periodically close and reopen the alternative log data set so that you can view the log records.

If you specify VIEW=Y on z/OS, the following considerations apply:

- If you use a GDG for alternative logging on z/OS, you must specify GDG(0) and DISP=SHR in a single DTLLOGnn DD statement in the PowerExchange Listener JCL. Also, allocate and create at least one generation of the GDG data set before starting the PowerExchange Listener.
- If you use third-party products that manipulate data set allocations, these products might interfere with VIEW=Y processing. For example, the products might change the SYSDSN ENQ to EXCLUSIVE mode, which prevents you from viewing the data set.
- The performance of the PowerExchange job that writes data to the alternative log data set might be degraded because of frequent data set open and close requests. Use the default value of 99 for the FLUSH parameter to minimize performance degradation.

Default is N.

Usage Notes:

- Use alternative logging to improve logging performance and to customize the amount of data logged for long-running jobs, such as a PowerExchange Logger for Linux, UNIX, and Windows process that runs in continuous mode.

- When dynamic alternative logging is enabled, PowerExchange creates a set of alternative log files for each PowerExchange process in a separate directory.

You can specify the location, the number of log files, and the log file size in megabytes. When a log file reaches the specified size, PowerExchange switches to the next log file and begins overwriting any data in that file.

- If you define the TRACING statement, also define the LOGPATH statement to specify a directory for the alternative log files on a Linux, UNIX, or Windows system.
- PowerExchange dynamically allocates the alternative log data sets unless you define DTLLOGnn DD statements in the JCL for a PowerExchange job or started task.
- On z/OS, Informatica recommends that you specify SYSOUT=* in a DTLLOG01 DD statement that you use in the JCL for all PowerExchange jobs and started tasks that issue messages, for example:

```
//DTLLOG01 DD SYSOUT=*
```

This strategy simplifies configuration because you define only one DTLLOG01 DD statement with a single SYSOUT option. Also, this strategy makes finding message output for a particular execution of a job or task easier because PowerExchange writes all of the message output to a single SYSOUT data set, which is available with the other job output.

- If you use a GDG on z/OS for the alternative logging, specify GDG(0) in the DTLLOGnn DD statement of the PowerExchange Listener JCL. For example:

```
DTLLOG01 DD DSN=USER1.V901.TRCGDG(0),DISP=SHR
```

By using GDG(0), you can use APPEND=Y to resume logging messages to the current generation. Also, you can use VIEW=Y to view log records in the GDG while the PowerExchange Listener task is active. If you specify APPEND=N, PowerExchange creates a new generation when the PowerExchange Listener starts.

If you use GDG(+1) instead, PowerExchange ignores the APPEND and VIEW parameters on the TRACING statement and creates a new generation whenever the PowerExchange Listener starts.

Also, when using a GDG, allocate and create at least one generation of the GDG before starting the PowerExchange Listener.

- On z/OS, if you use a GDG for alternative logging or specify a DTLLOG01 DD statement in the JCL for a PowerExchange job or started task, the FILENUM parameter is ignored.

GDG Example:

To append messages to the current generation of a GDG on z/OS, GDG(0), and be able to view the messages periodically, complete the following steps:

1. Allocate and create at least one generation data set in the GDG by running a batch job that contains JCL statements such as:

```
//DJEKGDG@ JOB (ACCOUNT),'GDG',NOTIFY=&SYSUID
//JSTEP01 EXEC PGM=IDCAMS
//SYSPRINT DD SYSOUT=*
//SYSIN DD *
DEFINE GDG-
(NAME(USER1.V901.TRCGDG)-
LIMIT(4)-
NOEMPTY-
SCRATCH)
//STEP2 EXEC PGM=IEFBR14
//DLLOG01 DD DSN=USER1.V901.TRCGDG(+1),DISP=(NEW,CATLG),
DCB=(BLKSIZE=32718,LRECL=132,RECFM=VB),
SPACE=(CYL,(1,1))
```

2. In the DBMOVER configuration member, define a TRACING statement that includes VIEW=Y and APPEND=Y. For example:

```
TRACING=(PFX=PWX,APPEND=Y,VIEW=Y)
```

This statement configures PowerExchange to append messages to the end of the current generation of the GDG and to periodically close and reopen the generation data set to make the messages available for viewing.

3. Configure the DTLLOG nn DD statement in the PowerExchange Listener JCL to reference the current generation data set of the GDG, for example:

```
DTLLOG01 DD DSN=USER1.V901.TRCGDG(0),DISP=SHR
```

Use DISP=SHR in this statement.

4. Start the PowerExchange Listener.

PowerExchange begins appending messages at the end of the current generation data set.

If the current generation does not exist, a JCL error occurs and the PowerExchange Listener does not start.

RELATED TOPICS:

- [“PowerExchange Message Logs” on page 218](#)
- [“Allocation Options for Alternative Log Data Sets on z/OS” on page 222](#)

UNIT Statement

The UNIT statement specifies the generic or esoteric unit name that PowerExchange uses when dynamically allocating target data sets on z/OS, if you do not provide UNIT information.

Operating Systems: z/OS

Related Statements: BS, DISP, LRECL, RECFM, RELEASE, SPACE, and VOLSER

Required: No

Syntax:

```
UNIT=unit
```

Value: For the *unit* variable, enter the generic or esoteric unit name that PowerExchange uses when dynamically allocating target data sets on z/OS, if you do not provide UNIT information. A valid value is a one- to eight-character alphanumeric string.

USE_DB_AUTH Statement

When LDAP authentication is enabled, the USE_DB_AUTH statement controls whether or not LDAP authentication applies to relational connections.

When LDAP authentication is not enabled through the SECURITY statement, this statement has no effect.

Operating Systems: Linux, UNIX, and Windows

Related Statements: SECURITY

Required: No

Syntax:

```
USE_DB_AUTH=Y|N
```

Valid Values:

- **N.** PowerExchange authenticates users against LDAP, regardless of the connection type.

- **Y.** For relational connections, PowerExchange does not authenticate users against LDAP. Instead, PowerExchange delegates authentication to an underlying relational database. For nonrelational connections, PowerExchange authenticates users against LDAP.

Default is Y.

USE_TYPE1_FALLBACKS Statement

The USE_TYPE1_FALLBACKS statement enables type 1 fallback mappings for all code pages. A type 1 fallback mapping is a one-way mapping from a Unicode character to a character in the output code page.

To enable type 1 fallback mappings, PowerExchange calls the `ucnv_setFallback()` API for all code pages that it uses.

Operating Systems: All

Related Statements: ICUCNVPROPERTY

Required: No

Syntax:

```
USE_TYPE1_FALLBACKS={N|Y}
```

Valid Values:

- **N.** Type 1 fallback mappings are not globally enabled.
- **Y.** Type 1 fallback mappings are enabled for all code pages.

Default is N.

Usage Notes: When PowerExchange converts data between Unicode and a source or target code page, type 1 fallback mappings are disabled by default. That is, when the code page conversion routine encounters a character mapping with a precision indicator of 1 in the UCM file, the mapping is not executed. Instead, the conversion routine generates a substitution output character.

To enable type 1 fallback mappings, define `USE_TYPE1_FALLBACKS=Y` in the DBMOVER configuration file. Alternatively, you can enable type 1 fallback mappings for a specific code page by using the `USE_TYPE1_FALLBACKS` option of the `ICUCNVPROPERTY` statement in the DBMOVER file.

In a PowerExchange session, a code page conversion routine might use type 1 fallback mappings when writing data to a target. Type 1 fallback mappings are not typically used when reading data from a PowerExchange source.

Informatica recommends that you enable type 1 fallback mappings whenever you use PowerExchange to write multibyte data to a target.

USESUP Statement

The USESUP statement controls whether PowerExchange uses zIIP for zIIP-enabled PowerExchange Listener functions.

Operating Systems: z/OS

Related Statements: SUP_FUNC, SUP_REQUEST, SUP_SSNAME, SUP_SSTYPE, WORKCLASS

Data Sources: All

Required: No

Syntax:

```
USESUP={N|Y}
```

Valid Values:

- **N.** PowerExchange does not use zIIP for zIIP-enabled PowerExchange Listener functions.
- **Y.** PowerExchange uses zIIP for zIIP-enabled PowerExchange Listener functions.

Default is N.

For more information about offloading PowerExchange processing to a zIIP, see the following guides:

- *PowerExchange Bulk Data Movement Guide*
- *PowerExchange CDC Guide for z/OS*

VOLSER Statement

The VOLSER statement specifies the volume serial number that PowerExchange uses when dynamically allocating target data sets on z/OS, if you do not provide VOLSER information.

Operating Systems: z/OS

Related Statements: BS, DISP, LRECL, RECFM, RELEASE, SPACE, and UNIT

Required: No

Syntax:

```
VOLSER=volume_serial
```

Value: For the *volume_serial* variable, enter the volume serial number that PowerExchange uses when dynamically allocating target data sets on z/OS, if you do not provide VOLSER information. A valid value is a one- to eight-character alphanumeric string.

VSAM Statement

The VSAM statement specifies the number of buffers that PowerExchange uses for data and index control intervals when processing VSAM data sets.

For more information about optimizing performance and system-managed buffering, see the DFSMS documentation.

Operating Systems: z/OS

Data Sources: VSAM data sets

Required: No

Syntax:

```
VSAM=( [BUFNI=index_buffers]  
      [,BUFND=data_buffers]  
      )
```

Parameters:

BUFNI=*index_buffers*

Optional. Number of I/O buffers that PowerExchange uses for VSAM index control intervals. A buffer is the size of a control interval in the index.

For the *index_buffers* variable, enter one of the following values:

- 0, to use the operating system default
- A number from 1 through 255

BUFND=*data_buffers*

Optional. Number of I/O buffers that PowerExchange uses for VSAM data control intervals. A buffer is the size of a control interval in the data component.

For the *data_buffers* variable, enter one of the following values:

- 0, to use the operating system default
- A number from 2 through 255

Note: You cannot specify 1 for BUFND.

Usage Notes:

- Additional index buffers improve performance by providing for the residency of some or all of the high-level index. Additional buffers minimize the number of high-level index records retrieved from DASD for key-direct processing.
Tip: The usual practice is to set the number of buffers that PowerExchange uses for data and index control intervals in data maps for individual bulk data movement operations rather than to use the global VSAM statement to set these values.
- Additional data buffers benefit direct inserts or updates during control area splits and spanned record processing.
- The maximum number of buffers allowed is 255, which represents 254 data buffers and one insert buffer.

WAITDSN Statement

The WAITDSN statement controls whether netport jobs wait for in-use data sets.

Operating Systems: z/OS

Related Statements: GDGLOCATE, NETPORT, and TAPEWAIT

Required: No

Syntax:

```
WAITDSN=Y
```

Value: Enter Y. PowerExchange waits for in-use data sets by specifying the S99WTDSN bit in the S99FLAG2 indicators on dynamic allocation (SVC99) requests.

Usage Notes:

- By default, PowerExchange does not wait for in-use data sets.
- To use this statement, you must APF-authorize the libraries in the STEPLIB of the netport JCL.
- PowerExchange ignores this statement when initializing the PowerExchange Listener.

WORKCLASS Statement

Defines the transaction name for Workload Manager classification when PowerExchange zIIP exploitation is enabled.

Operating Systems: z/OS

Related Statements: SUP_FUNC, SUP_REQUEST, SUP_SSNAME, SUP_SSTYPE, USESUP

Data Sources: All

Required: No

Syntax:

```
WORKCLASS=transaction_name
```

Value: Enter a name of up to eight characters.

Default is PWXWORK.

For more information about offloading PowerExchange processing to a zIIP, see the following guides:

- *PowerExchange Bulk Data Movement Guide*
- *PowerExchange CDC Guide for z/OS*

WRT_ERROR_HANDLING Statement

The WRT_ERROR_HANDLING statement indicates whether to use an alternative method of handling error messages returned from the PowerExchange Listener for bulk data movement sessions that use writer partitioning when the **Write Mode** connection attribute is set to **Confirm Write On**.

Operating System: Linux, UNIX, or Windows, wherever the PowerCenter Integration Service runs

Data Sources: All

Data Targets: VSAM data sets and sequential files

Required: No

Syntax:

```
WRT_ERROR_HANDLING={N|Y}
```

Valid Values:

- **N.** Use the default method of handling error messages returned from the PowerExchange Listener on the target.
- **Y.** Use the alternative method of handling these error messages. The alternative method can be more efficient in a particular situation. Specify Y only under the following conditions:
 - Writer partitioning is enabled.
 - The **Write Mode** connection attribute is set to **Confirm Write On**.
 - The input data for the writer partitions contains many errors.

Default is N.

Usage Notes: In PowerCenter, you can override this WRT_ERROR_HANDLING setting for a bulk data movement session that has writer partitions. Enter the WRT_ERROR_HANDLING setting in the **PWX Override** attribute on the PWX NRDB Batch application connection. Use the same syntax as for this DBMOVER statement.

If you specify WRT_ERROR_HANDLING=Y and set the **Write Mode** connection to **Confirm Write Off**, this statement is ignored.

Consumer API (CAPI) Connection Statements

CAPI connection statements configure how PowerExchange captures and extracts change data from the change stream for a database instance.

Specify CAPI connection statements in the DBMOVER configuration file of the PowerExchange installation that processes the change stream. If you offload extraction processing, some additional configuration considerations apply.

For more information, see the *PowerExchange CDC Guide for Linux, UNIX, and Windows* and *PowerExchange CDC Guide for z/OS*.

You must define at least one source-specific CAPI_CONNECTION statement for each source type. For z/OS sources and for Oracle, you must also specify the UOWC CAPI_CONNECTION statement for the UOW Cleanser.

If required, specify multiple CAPI_CONNECTION statements of the same type or of different types in the same DBMOVER file.

Multiple CAPI Connections and Overrides

PowerExchange can capture changes for more than one source type using a single PowerExchange Listener and DBMOVER configuration file on a single server.

You can define a maximum of eight CAPI_CONNECTION statements in a DBMOVER configuration file.

The CAPI_CONNECTION statements can be of the same type or of mixed types, as indicated by the TYPE parameter. The NAME parameter in each statement must specify a unique name.

If you define multiple CAPI_CONNECTION statements for a source type, you can optionally define a CAPI_SRC_DFLT statement to identify the default CAPI_CONNECTION for that source type. Also, you can optionally specify a CAPI_CONN_NAME parameter that specifies an overall default statement, out of all of the CAPI_CONNECTION statements defined in the DBMOVER file.

Instead of specifying defaults, you can use the following CAPI connection name overrides to point to a specific CAPI_CONNECTION statement:

- For a CDC session, the **CAPI Connection Name Override** attribute in the PowerCenter PWX application connection.
- For CAPXRT database row tests in the PowerExchange Navigator, the **CAPI Connection Name** value in the **Advanced CAPXRT Parameters** dialog box. If you add an SQL statement for generating restart tokens, include the CONNAME parameter to point to the override CAPI_CONNECTION.
- If you use the DTLUAPPL to generate restart tokens, the CONN_OVR parameter in the DTLUAPPL control statement.
- For the PowerExchange Logger for Linux, UNIX, and Windows, the CONN_OVR parameter in pwxcl.cfg configuration file.
- For PowerExchange ODBC connections, the DTLCONN_OVR parameter in the odbc.ini file or the SQL escape sequence override DTLCONNOVR.

Informatica recommends that you use the following overrides:

- If you extract change data from Linux, UNIX, and Windows sources, specify a **CAPI Connection Name Override** value in the application connections for your CDC sessions.
- If you use the PowerExchange Logger for Linux, UNIX, and Windows, specify the CONN_OVR parameter in the pwxcl.cfg file.

The following example statements include multiple source-specific CAPI_CONNECTION statements for DB2 and Oracle, with an overall CAPI_CONN_NAME default and a CAPI_SRC_DFLT default for DB2:

```
CAPI_CONN_NAME=DTLUDB1
/*
CAPI_CONNECTION=(NAME=DTLUDB1,TYPE=(UDB,CAPINAME=CAPIUDB1))
CAPI_CONNECTION=(NAME=CAPIUDB1,TYPE=(UDB,
DATABASE=SAMPLE1,DBCONN=SAMPLE1,PASSWORD=user,USERID=password,
CCATALOG=DTLCCATLAOG))
/*
CAPI_CONNECTION=(NAME=DTLUDB2,TYPE=(UDB,CAPINAME=CAPIUDB2))
CAPI_CONNECTION=(NAME=CAPIUDB2,TYPE=(UDB,
DATABASE=SAMPLE2,DBCONN=SAMPLE2,PASSWORD=password,USERID=user,
CCATALOG=DTLCCATLAOG))
/*
CAPI_CONNECTION=(NAME=DTLUDB,TYPE=(UDB,CAPINAME=CAPIUDB,))
CAPI_CONNECTION=(NAME=CAPIUDB,TYPE=(UDB,
DATABASE=SAMPLE,DBCONN=SAMPLE,PASSWORD=password,USERID=user,
CCATALOG=DTLCCATALOG))
CAPI_SRC_DFLT=(UDB,DTLUDB2)
/*
ORACLEID=(OEMCAP,OEMDB,OEMDB.INFORMATICA.COM,OEMDB.INFORMATICA.COM)
CAPI_CONNECTION=(NAME=CAPIUOWC,TYPE=(UOWC,CAPINAME=CAPIORA))
CAPI_CONNECTION=(NAME=CAPIORA,TYPE=(ORCL,ORACOLL=OEMCAP))
```

Order of Precedence for CAPI Statements and Overrides

PowerExchange uses the following order of precedence when determining which CAPI connection information to use:

1. CAPI connection name overrides take precedence over the CAPI_SRC_DFLT and CAPI_CONN_NAME statements.
2. A CAPI_SRC_DFLT statement for a specific source type takes precedence over a CAPI_CONN_NAME statement for that source type.
3. If you do not specify a CAPI connection name override or a CAPI_SRC_DFLT statement, PowerExchange uses the CAPI_CONN_NAME statement.
4. If you do not specify a CAPI connection name override or CAPI_SRC_DFLT statement for the source, and a CAPI_CONN_NAME statement is also not available, PowerExchange uses the first CAPI_CONNECTION statement for the source type in the DBMOVER configuration file.

Tip: Informatica recommends that you specify a CAPI connection name override for the most efficient CAPI processing.

Source-Specific CAPI_CONNECTION Statements

All PowerExchange CDC systems require CAPI_CONNECTION statements for capture and extraction processing.

You define these statements in the DBMOVER configuration file. The types of CAPI_CONNECTION statements that you define vary by source type and system.

The following table identifies the required and optional CAPI_CONNECTION statements types by operating system:

Source System	CAPI_CONNECTION Types
All z/OS sources	<ul style="list-style-type: none"> - LRAP CAPI_CONNECTION for the Log Read API that extracts change data from PowerExchange Logger for z/OS log files (Required) - UOWC CAPI_CONNECTION for the UOW Cleanser (Required)
DB2 for i5/OS	<ul style="list-style-type: none"> - AS4J CAPI_CONNECTION for the journal reader that extracts change data from DB2 journals (Required) - UOWC CAPI_CONNECTION for the UOW Cleanser (Required)
DB2 for Linux, UNIX, and Windows	<ul style="list-style-type: none"> - UDB CAPI_CONNECTION for change extraction from DB2 recovery logs (Required) - CAPX CAPI_CONNECTION, if you use continuous extraction mode (Optional)
Microsoft SQL Server	<ul style="list-style-type: none"> - MSQJ CAPI_CONNECTION for change extraction from Microsoft SQL Server distribution databases (Required) - CAPX CAPI_CONNECTION, if you use continuous extraction mode (Optional)
Oracle	<ul style="list-style-type: none"> - ORAD CAPI_CONNECTION for change extraction from Oracle redo logs (Required) - CAPX CAPI_CONNECTION, if you use continuous extraction mode (Optional)

DBMOVER Statements for Commands Issued through the pwxcmd Program

To send pwxcmd commands to a PowerExchange process that is not managed by an application service, configure the PowerExchange process to receive pwxcmd commands. Also, on the Linux, UNIX, or Windows system from which you issue pwxcmd commands, configure a connection to the PowerExchange process.

To configure a PowerExchange Listener process to receive pwxcmd commands, define DBMOVER statements on the node where the PowerExchange Listener process runs.

The following table describes these statements:

Statement	Description	Reference
LISTENER	Required. Defines the TCP/IP port on which a named PowerExchange Listener process listens for work requests.	“LISTENER Statement” on page 142
SECURITY	Optional. Controls whether PowerExchange authenticates user access to PowerExchange and authorizes users to issue specific pwxcmd commands.	“SECURITY Statement” on page 171
SVCNODE	Required. Specifies the TCP/IP port on which a PowerExchange Listener process listens for commands.	“SVCNODE Statement” on page 186

To configure a PowerExchange log-based ECCR on z/OS to receive pwxcmd commands, define a DBMOVER SVCNODE statement on the node where the PowerExchange ECCR job or started task runs.

The following table describes the statement:

Statement	Description	Reference
SVCNODE	Required. Specifies the ECCR name and TCP/IP port on which the ECCR listens for commands.	“SVCNODE Statement” on page 186

To configure a PowerExchange Logger for Linux, UNIX, and Windows process to receive pwxcmd commands, define configuration statements on the node where the PowerExchange Logger process runs.

The following table defines these statements:

Statement	Configuration File	Description	Reference
CONDENSENAME	pwxcl.cfg file	Required. Defines a name for the command-handling service for a PowerExchange Logger process that is the target of pwxcmd commands.	CDC guide for the operating system
SECURITY	DBMOVER configuration file	Optional. Controls whether PowerExchange authenticates user access to PowerExchange and authorizes users to issue specific pwxcmd commands.	“SECURITY Statement” on page 171
SVCNODE	DBMOVER configuration file	Required. Specifies the TCP/IP port on which a PowerExchange Logger for Linux, UNIX, and Windows process listens for commands.	“SVCNODE Statement” on page 186

Finally, configure a connection to the PowerExchange process to which you want to send pwxcmd commands.

The following table describes the DBMOVER statement that you define on the Linux, UNIX, or Windows system from which you issue pwxcmd commands:

Statement	Configuration File	Description	Reference
CMDNODE	dbmover.cfg file	Required. Specifies connection information for a PowerExchange process that is the target of pwxcmd commands.	“CMDNODE Statement” on page 97

Sample DBMOVER Configuration Files

PowerExchange provides example DBMOVER configuration files. Use the example files as a starting point when creating your configuration files.

The example files are available at the following locations:

- On i5/OS, the DBMOVER member in the CFG file in the *datilib* library

- On Linux, UNIX, and Windows, the dbmover.cfg or dbmover.cfg_sample file in the PowerExchange installation directory
- On z/OS, the DBMOVER member in the RUNLIB library

Note: The z/OS DBMOVER member includes variables in pointed brackets < >, which PowerExchange replaces with specific values when you use the z/OS Installation Assistant to complete the installation.

On Linux, UNIX, and Windows, full and upgrade installations of PowerExchange provide the dbmover.cfg file. Hotfix installations provide the dbmover.cfg_sample file if the sample file has changed from the previous release.

Also, on Linux, UNIX, and Windows, PowerExchange provides sample configuration files for use in localized environments.

Full and upgrade installations of PowerExchange on Linux, UNIX, and Windows provide the following sample files:

File Name	Environment
dbmover.cfg	English
dbmover_bz.cfg	Brazilian Portuguese
dbmover_ja.cfg	Japanese
dbmover_ko.cfg	Korean
dbmover_sc.cfg	Simplified Chinese

PowerExchange hotfix installations on Linux, UNIX, and Windows provide the following sample files as needed:

File Name	Environment
dbmover.cfg_sample	English
dbmover_bz.cfg_sample	Brazilian Portuguese
dbmover_ja.cfg_sample	Japanese
dbmover_ko.cfg_sample	Korean
dbmover_sc.cfg_sample	Simplified Chinese

CHAPTER 3

Netport Jobs

This chapter includes the following topics:

- [Netport Jobs Overview, 209](#)
- [Configuring Netport Jobs, 210](#)
- [Sample Netport Jobs, 210](#)
- [Job Status Messages for Netport Jobs, 211](#)
- [PowerExchange Substitution Variables for Netport Jobs, 212](#)
- [Netport Jobs and Offload Processing, 216](#)
- [Netport Jobs and DB2 Bulk Data Loads, 216](#)
- [Using Netport Jobs for Generation Data Sets, 216](#)

Netport Jobs Overview

You can associate JCL streams, called *netport jobs*, to specific ports to access a source or target in certain situations.

Use a netport job in the following situations:

- Access a source or target through a non-PowerExchange module such as IMS DLIBATCH.
For IMS access, the IMS ODBA access method is not supported with netport jobs.
- Avoid excessive wait times during session processing. For example, if the data is on tape, use a netport job to avoid significant delays that result from tape mounts.
- Process a generation data group (GDG). In this case, special considerations and configuration requirements apply. See [“Using Netport Jobs for Generation Data Sets” on page 216](#).
- Access IDMS data when full user ID checking is required, as when the SECURITY statement is set to (2,x) in the DBMOVER configuration file.

To link a netport job to a port, you must define a pair of NETPORT and LISTENER statements in the DBMOVER configuration file.

RELATED TOPICS:

- [“Using Netport Jobs for Generation Data Sets” on page 216](#)

Configuring Netport Jobs

Configure a netport job if your situation requires one. To use DL/I or BMP access to IMS data, you must configure a netport job.

The DBMOVER configuration file configures a netport job for tapes and GDGs and a netport job for IMS unless IMS is not required. The following statements are included at installation for an IMS netport job:

```
LISTENER=(node1,TCPIP,12480)
NETPORT=(node1,12480,,, "DTLUSR.V800B11.RUNLIB (IMSJCL) ", PSB1)
```

For a new netport job, you must add a LISTENER statement and associated NETPORT statement in the DBMOVER member. You can use the LISTENER and NETPORT statements for the TAPEJCL member as a template. In the NETPORT statement, assign a port to the netport job and retain the positional commas. Verify that the first parameter in the LISTENER and NETPORT statements point to the node name in the LISTENER statement that specifies the port on which this PowerExchange Listener is listening. The default configuration uses node "node1" and port "2480."

1. Configure the netport JCL. Use the sample JCL members.

Note: In the netport JCL, you can include substitution variables instead of specific values. For example, if you include the PSB=%PSBNAME variable instead of a specific PSB name in the IMSJCL member, you can override the PSB name when you perform a PowerExchange Navigator database row test or run a PowerCenter session. By using the substitution variable with an override, you can use the same JCL for multiple PSBs. For more information about substitution variables, see [“PowerExchange Substitution Variables for Netport Jobs” on page 212](#).

2. Add a LISTENER statement in the DBMOVER member that defines a port for listening for netport job requests.
3. Add a NETPORT statement in the DBMOVER member that associates the data set that contains the netport JCL to the listener port.
4. In the PowerExchange Navigator, run a row test to read data.

You must add a NODE statement in the dbmover.cfg file on Windows. For example:

```
NODE=(imsnet,TCPIP,12.34.56.78,12480)
```

When running a row test against an IMS data map, use the node name to access the data. The PowerExchange Listener passes the PSB name from the NETPORT statement to the JCL as a parameter and initiates submission of the IMSJCL job.

Sample Netport Jobs

PowerExchange provides sample netport jobs in the RUNLIB library. Customize the job that best fits your situation.

The following members contain sample JCL:

- CAPXJCL, for CDC data sources

- IMSJCL, for IMS bulk data sources
- GDGJCL and TAPEJCL, for bulk data sources other than IMS
- NETJCL, for examples of substitution variables

Note: The CAPXJCL, GDGJCL, and TAPEJCL members run the PowerExchange Listener for netport jobs, DTLLST3. The IMSJCL member runs an IMS PROC, DLIBATCH. The NETJCL runs IEFBR14.

If you need other netport jobs, create them by using the JCL of one of the sample members as a template.

The sample members include REGION statements that allocate 64 MB of memory for non-IMS jobs or 128 MB of memory for IMS jobs. Verify that the JCL for netport jobs allocates sufficient memory for the PowerExchange processing in your environment. Memory usage is a particular concern if PowerExchange must perform an ICU-based code page conversion.

If IDMS access is required and the SECURITY parameter in the DBMOVER configuration file is set to 2 (each user who submits an IDMS access job must have their user ID and password checked by the relevant security package), you must use a netport job. Copy the TAPEJCL member under a new and relevant name. Make sure that the PowerExchange copies of the IDMS.LOADLIB and IDMS.DBA.LOADLIB libraries are in the STEPLIB. Code the following DD statement in the JCL that you created:

```
//SYSIDMS      DD DSN=&HLQ..RUNLIB(DTLDML),
//              DISP=(SHR)
```

Ensure that the relevant SYSCTL statement is included in the JCL for Central Versions, or the following DD statements if running local:

```
//IDMSDCT      INCLUDE MEMBER=IDMSDICT
//IDMSFIL      INCLUDE MEMBER=IDMSFILE
```

These members will need the relevant dictionary definitions and database file definitions.

Note: The PowerExchange Listener must be restarted to detect a new netport job.

RELATED TOPICS:

- [“Configuring Netport Jobs” on page 210](#)

Job Status Messages for Netport Jobs

PowerExchange writes informational messages, including netport job status messages, to the PowerExchange message log file. You can also configure PWXPC to write informational messages to the PowerCenter session log. The session log then contains all of the PowerCenter and PowerExchange messages related to session processing.

PowerExchange writes the following message to the PowerExchange log file when a netport job begins:

```
PWX-00178: Process started: Jobname=jobname,Jobid=jobid,User=user_ID,
          tablename_or_mapname,file_name
```

PowerExchange writes the following message to the PowerExchange log file when a netport job ends:

```
PWX-00179: Process ended: Jobname=jobname,Jobid=jobid,User=user_ID,Highest Rc=max_rc
```

PWXPC incorporates these and other PowerExchange messages into PowerCenter message PWXPC_10091 when writing messages to the session log.

To configure PWXPC to include PowerExchange informational messages in the PowerCenter session log, configure one of the following connection attributes in PowerCenter:

- Select the **Retrieve PWX Log Entries** connection attribute. (PWXPC application connection types only)
- Specify the RETLOGINFOMSG=Y option in the **PWX Override** connection attribute. (All PWXPC connection types)

For more information, see *PowerExchange Interfaces for PowerCenter*.

PowerExchange Substitution Variables for Netport Jobs

To control and customize netport jobs, PowerExchange provides substitution variables. Prior to submitting a netport job to z/OS, PowerExchange resolves the defined substitution variables in the skeleton netport JCL with the appropriate values.

The skeleton JCL member is identified in the fifth positional parameter of the NETPORT statement in the DBMOVER configuration file.

The sample JCL in the NETJCL member of the RUNLIB library demonstrates how to enter substitution variables to form unique data set names. You can include the variables in a netport job that is based on the sample CAPXJCL, GDGJCL, IMSJCL, or TAPEJCL JCL.

The following table describes the substitution variables:

Substitution Variable	Where Used in the JCL	Description
%CLIENT	Fourth positional parameter on the first line of the PARMS statement	Unique 16-byte hexadecimal identifier that includes the job name of the PowerExchange Listener.
%CPNODE	First positional parameter on the first line of the PARMS statement	The <i>cpnode</i> parameter value from the NETPORT statement that PowerExchange used to submit the netport job.
%CPPORT	Third positional parameter on the first line of the PARMS statement	The <i>cpport</i> parameter value from the NETPORT statement that PowerExchange used to submit the netport job.
%DATE_YYMMDD	DD statement	Current Gregorian date.
%DATE_YYYYDDD	DD statement	Current Julian date.

Substitution Variable	Where Used in the JCL	Description
%DMX_ECSA	Keyword parameter on the second line of the PARMS statement	<p>The hexadecimal address of the ECSA memory that holds the time of the latest update to the DATAMAPS file.</p> <p>Use the DMX_ECSA keyword parameter to enter this substitution variable on the second line in the PARMS DD, as follows:</p> <pre>DMX_ECSA=%DMX_ECSA</pre> <p>This parameter is optional. Include it if you use data maps caching and want to enable a netport job to initialize quickly in an environment where APF authorization is not allowed. For example, use this parameter an environment where access to IMS databases is through DL/I.</p> <p>If you do not use data maps caching, this parameter is ignored.</p>
%IMSID	-	<p>A substitution variable for the IMS SSID.</p> <p>if you specify an IMS SSID override in the PowerCenter IMS SSID Override session property for a source or target or in the IMS SSID advanced parameter for a database row test, the override value replaces the substitution variable for the session or row test.</p> <p>If you do not specify an IMS SSID override but you use an ODBA data map with a BMP netport job, the IMS SSID from the data map replaces the substitution variable.</p> <p>If an IMS SSID is not available from the data map or entered as an IMS SSID override, the substitution variable is replaced by a NULL value. In this case, IMS determines which IMS subsystem to access based on the load libraries specified in the netport job.</p>
%Nn	JOB statement	<p>Value of the numeric counter with the length of <i>n</i>. Valid values for the length are from 1 through 7.</p> <p>The PowerExchange Listener maintains a separate value for each possible numeric counter, and increases the value by one each time that numeric counter is used. For example, if netport JCL specifies %N3 in two places in the JCL, the first occurrence is assigned a value of 001 and the second occurrence is assigned a value of 002. The next netport job that references %N3 is assigned 003, and so on.</p>

Substitution Variable	Where Used in the JCL	Description
%PSBNAME	-	<p>A substitution variable for the PSB name.</p> <p>If you specify a PSB name override in the PowerCenter IMS PSBNAME Override session property for a source or target or in the PowerExchange Navigator PSB Name advanced parameter for a database row test, the override value replaces the substitution variable for the session or row test. By using the substitution variable with an override, you can use the same JCL and same set of NETPORT and LISTENER statements for multiple PSBs. You do not need to edit the DBMOVER statements and restart the PowerExchange Listener.</p> <p>If you do not specify a PSB name override but you use an ODBA data map with a DL/I or BMP access method override, this substitution variable is replaced by the PSB name from the data map.</p> <p>Note: This substitution variable has no effect on the %1 variable in the NETPORT statement. This variable gets the PSB name from the sixth positional parameter in the NETPORT statement.</p> <p>Also, in the netport JCL, you can allocate a database member to the netport job, which is used for accessing the required database data sets. If this database member has the same name as the PSB, you can add an INCLUDE statement in the JCL that uses the %PSBNAME variable. For example:</p> <pre>//DBALLOC INCLUDE MEMBER=%PSBNAME</pre> <p>By using this variable with PSB overrides, you can allocate other database members to the job.</p>
%PWD	JOB statement	The password of the user ID that connected to the PowerExchange Listener. To avoid exposing passwords, you should only use this substitution variable on the PASSWORD parameter of the JOB card.
%SOCKNO	Second positional parameter on the first line of the PARMS statement	Socket number on which the connection to the PowerExchange Listener was made.
%SMFTASK	-	If the STATS statement in the DBMOVER member specifies recording to SMF, this variable is the PowerExchange task number. Otherwise, the value is 00000.
%TIME_HHMMSS	DD statement	Current time, in 24-hour clock time format.
%USER	JOB statement	The user ID that connected to the PowerExchange Listener.
%1	-	An optional substitution variable that the sample IMSJCL member of the RUNLIB library uses to populate the PSB parameter in the DLIBATCH PROC. Enter this variable as the sixth positional parameter in the NETPORT statement in the DBMOVER member.
%2	-	An optional substitution variable. Enter it as the seventh positional parameter in the NETPORT statement in the DBMOVER member.
%3	-	An optional substitution variable. Enter it as the eighth positional parameter in the NETPORT statement in the DBMOVER member.

Substitution Variable	Where Used in the JCL	Description
%4	-	An optional substitution variable. Enter it as the ninth positional parameter in the NETPORT statement in the DBMOVER member.
%5	-	An optional substitution variable. Enter it as the tenth positional parameter in the NETPORT statement in the DBMOVER member.
%6	-	An optional substitution variable. Enter it as the eleventh positional parameter in the NETPORT statement in the DBMOVER member.
%7	-	An optional substitution variable. Enter it as the twelfth positional parameter in the NETPORT statement in the DBMOVER member.
%8	-	An optional substitution variable. Enter it as the thirteenth positional parameter in the NETPORT statement in the DBMOVER member.

Example of IMS SSID and PSB Substitution for an IMS Netport Job

This example demonstrates how to use the %PSBNAME substitution variable in the netport JCL to substitute in PSB name values for a bulk data movement session.

This example uses the following assumptions:

- The source data map that was imported into PowerCenter uses the DL/1 BATCH access method. It does not specify a PSB name.
- The IMSJCL member for the netport job includes the %PSBNAME variable:

```
//STEP1 EXEC PROC=IMSBATCH,NBA=5,OBA=5,
//      IMSID=%IMSID,
//      MBR=DTLLST3I,REGION=32M,PSB=%PSBNAME
```

- The DBMOVER member contains a NETPORT statement that includes a specific PSB name:

```
NETPORT=(NODE1,26580,,,"PWX.PROD1.RUNLIB(IMSJCL)",DTLPB05)
```

When the netport job runs, PowerExchange replaces the %PSBNAME variable in the JCL with "DTLPB05" from the NETPORT statement.

Note: If you use an ODBA data map with an access method override, the %PSBNAME variable is replaced by the value from the data map. If you specify a PSB name override when you perform a database row test or run a PowerCenter session, the override takes precedence over the value in the NETPORT statement and in the data map.

You now need to use the PSB name of "DTLB06" but cannot define another NETPORT statement in the DBMOVER member because the maximum limit of ten statements has been reached. Instead, you define the **IMS PSBNAME Override** attribute in the PowerCenter session-level properties for the source. The netport job then uses the override PSB name instead of the PSB name from the NETPORT statement. When you use session-level overrides, you do not need to edit the DBMOVER member and restart the PowerExchange Listener.

After migrating your test environment to production, you want to execute the same netport PROC on the production system. In this case, you define the **IMS SSID Override** attribute in the PowerCenter session-level properties for the source.

Netport Jobs and Offload Processing

If you are using netport jobs and offload processing together, make sure that the PowerExchange Listener and the netport jobs point to the same data map file. This data map is specified using the //DATAMAP DD statements in the JCL of the PowerExchange Listener and the netport jobs.

Netport Jobs and DB2 Bulk Data Loads

When a DB2 bulk data load operation is run from a netport job, the load runs as a task of the netport job, not as a separately submitted job.

A sample netport job, such as CAPXJCL, can be used as a template for creating such a netport job. The sample job must be edited to add the DD statements that would normally be in the DB2 LOAD JCL.

For example:

```
//SORTOUT DD UNIT=SYSDA,SPACE=(4000,(20,20),,,ROUND)
//SORTWK01 DD UNIT=SYSDA,SPACE=(4000,(20,20),,,ROUND)
//SYSDISC DD UNIT=SYSDA,SPACE=(4000,(20,20),,,ROUND)
//SYSERR DD UNIT=SYSDA,SPACE=(4000,(20,20),,,ROUND)
//SYSMAP DD UNIT=SYSDA,SPACE=(4000,(20,20),,,ROUND)
//SYSUT1 DD UNIT=SYSDA,SPACE=(4000,(20,20),,,ROUND)
//UTPRINT DD SYSOUT=*
```

Note: The netport job cannot have a DD statement for SYSPRINT. This output is allocated to a file for DB2 LOAD.

Using Netport Jobs for Generation Data Sets

Generation Data Groups (GDGs) on z/OS offer a unique and versatile approach to accessing data. You can use a relative generation number to access the current generation data set in the GDG while maintaining a fixed data set name in the job JCL.

For example, if you use a GDG data set name in a data map, you can write to the generation data set by using the relative generation number. You do not need the data set name in the data map to be dynamically updated for each generation.

By default, the PowerExchange Listener refreshes the generation table for a GDG with the latest information from the z/OS catalog so that you can use a relative generation number, such as AAA.BBB.CCC(0), to access the latest generation. This behavior is appropriate for long running batch jobs and started tasks such as the PowerExchange Listener, which need to access the current generation by using a relative generation number.

Note: If you set the GDGLOCATE=N statement in the DBMOVER configuration member, after the PowerExchange Listener uses a relative generation number to access a generation data set the first time, all future Listener references to that GDG access the same generation. As a result, for the life of the batch job, the Listener accesses the same generation data set regardless of whether more current generations exist.

If you use relative generation references and want the PowerExchange Listener to refresh the generation table, perform one of the following actions:

- Use the default setting of GDGLOCATE=Y in the DBMOVER configuration member. This setting ensures that PowerExchange references the z/OS catalog to get the latest generation information when accessing a GDG to read an existing data set or create a new data set. With this setting, you do not need to restart the PowerExchange Listener.
- If you set GDGLOCATE=N in the DBMOVER configuration member, set up a netport job that allows the latest generation data set in a GDG to be referenced. This method is similar to how IMS is handled. Each time the file is accessed, the netport job shuts down and then another netport job is invoked. All GDGs are recognized, as intended.

Ensure that the netport job runs on the same z/OS image as the PowerExchange Listener to satisfy socket API call constraints. Otherwise, the job might timeout.

Setting Up Netport Jobs for GDGs

To set up a netport job for a GDG, complete the following steps.

1. Determine an available z/OS port number to use with the netport job.

In the example syntax in the following steps, this port number is "7777."

2. Add a NODE statement to your source and target DBMOVER configuration files. For example:

```
NODE=(MVS,TCPIP,mvs1,2480)
NODE=(MVSGDG,TCPIP,mvs1,7777)
```

If you use ODBC drivers to point from partner software, you need to add a driver for this new location.

3. Update the DBMOVER configuration file in the RUNLIB library on z/OS:

Add a LISTENER statement:

```
LISTENER=(node1,TCPIP,2480)
LISTENER=(node1,TCPIP,7777)
```

To indicate to the z/OS PowerExchange Listener that the second port is associated with a netport, add a NETPORT statement with the same port number. For example:

```
NETPORT=(node1,7777,,,"INFA.Vxyz.RUNLIB(GDGJCL)",,,,,,,)
```

4. Create the GDGJCL member in the RUNLIB library.

You can use the TAPEJCL member as a template. Add a job card with "%N5" appended to it, and any other DDs that you might need for your PowerExchange Listener job.

When the z/OS PowerExchange Listener detects activity on the netport, instead of directly trying to service it, the Listener tries to submit the JCL file that is specified in the fifth positional parameter of the NETPORT statement to the JES internal reader. As the JES reader reads and submits the JCL, it scans every JCL line for %x tokens, where x is a numeric value from 1 through 8. The JES reader substitutes these tokens with the appropriate values from parameters 6 through 13 in the NETPORT statement.

Tip: Do not confuse the %x tokens with the %Nx tokens. The %Nx tokens are used to generate incremented numeric values of x length.

You can then submit the GDGJCL job whenever a request is received through the assigned port for the netport job. This action enables a job to end and another job to begin and process the GDGs on z/OS appropriately.

5. Stop and restart the PowerExchange Listener on z/OS.

Whenever you access a GDG, point to the z/OS GDG location instead of to the z/OS location.

CHAPTER 4

PowerExchange Message Logs and Destination Overrides

This chapter includes the following topics:

- [PowerExchange Message Logs, 218](#)
- [PowerExchange Message Destination Overrides, 223](#)

PowerExchange Message Logs

PowerExchange stores informational, warning, and error messages about bulk data movement and change data capture (CDC) operations in PowerExchange message log files. PowerExchange also writes some message output to standard output (stdout). On z/OS and i5/OS systems, PowerExchange writes WTO messages to the system operator console.

Note: The text for the majority of PowerExchange messages is stored in the DTLMSG file, which is created during installation.

You can use the following types of message log files as the primary location for writing messages:

- **PowerExchange message log file.** PowerExchange writes all runtime messages to the PowerExchange message log file by default. PowerExchange creates this log file if it does not exist. PowerExchange writes each message, as it occurs, to this single log file. If the message log file becomes full, you must clear it.

If you enable alternative logging of messages, this primary message log file contains only the DBMOVER statements and any trace information.
- **PowerExchange alternative log files.** If you enable alternative logging by defining the TRACING statement in the DBMOVER configuration file, PowerExchange writes runtime messages from PowerExchange components, programs, and commands to a set of alternative message log files that are used on a rotating basis. PowerExchange dynamically allocates these alternative log files if they do not exist. When an alternative log file becomes full, PowerExchange switches to another alternative log file and begins overwriting any data in that file. Informatica recommends that you use alternative logging to prevent out-of-space conditions on log files, to improve logging performance, and to control the amount of data that is logged for long-running jobs. Also, with alternative logging, PowerExchange buffers messages and writes them to disk files based on a customized flush interval, thereby reducing I/O activity on these files.

The names of the primary message log file and the alternative log files vary by platform.

To redirect specific messages to destinations other than the default message log file, you can define a message destination override file, called DTLMSGO. The override destinations include the PowerExchange message log file or alternative log file, standard output (stdout), and i5/OS or z/OS system operator console.

RELATED TOPICS:

- [“TRACING Statement” on page 193](#)
- [“PowerExchange Message Destination Overrides” on page 223](#)

PowerExchange Message Log File

By default, PowerExchange writes most messages to the PowerExchange message log file.

PowerExchange creates this file if it does not exist.

The following table shows the default name and location of the PowerExchange message log file by operating system:

Operating System Type	Message Log File Name	Location
i5/OS	<i>datalib</i> /LOG(DTLLOG)	A member of the LOG file in the PowerExchange data library, <i>datalib</i> , which PowerExchange creates during installation.
Linux, UNIX, and Windows	detail.log	A file in the current working directory of the PowerExchange process that is running.
z/OS	DTLLOG DD data set	A data set that is allocated by the DTLLOG DD statement in the JCL for a PowerExchange component that issues messages. This DD statement can point to a permanent data set or to a SYSOUT data set.

When messages are written to a single file, without manual intervention, the volume of data continues to grow until the PowerExchange message log file becomes full. This out-of-space condition usually occurs for long-running PowerExchange tasks, such as the PowerExchange Listener. Also, PowerExchange must open and close the PowerExchange message log file to write each message. This open and close activity can decrease performance and increase resource utilization when message volume is high.

PowerExchange Alternative Log Files

PowerExchange can dynamically allocate a set of alternative log files for logging messages.

Use of PowerExchange alternative log files has the following benefits:

- PowerExchange uses multiple files for logging messages. When the current log file becomes full, PowerExchange can switch to the next log file, thereby preventing out-of-space conditions.
- After opening an alternative log file, PowerExchange buffers messages so that they can be written to a log file on disk at a specified interval. This buffering reduces the amount of open, close, and write activity on the file to improve performance and resource usage.
- You can control the size and number of alternative log files and the frequency with which PowerExchange flushes the log records to the file.

To configure PowerExchange to use alternative log files, define the TRACING statement in the DBMOVER configuration file. When alternative logging is enabled, PowerExchange writes runtime messages from PowerExchange components, programs, and commands to a set of alternative log files that are used on a rotating basis. By default, PowerExchange dynamically allocates five alternative log files.

Note: PowerExchange continues to write initial startup messages to the primary PowerExchange message log file.

On z/OS only, you can add DTLLOGnn DD statements in the JCL for a PowerExchange component that logs messages to alternative log data sets if you do not want to use the TRACING statement to dynamically allocate the alternative log data sets. Manual allocation by the DD statements overrides dynamic allocation. To send the message output to a JES2 or JES3 SYSOUT file rather than to a data set that you specify, enter a single DTLLOG01 DD statement in the JCL that specifies the SYSOUT parameter. By using SYSOUT, you can keep the output from a single PowerExchange Listener execution with the rest of the job output. If you use dynamic allocation, PowerExchange dynamically creates a set of log data sets in a separate directory for each PowerExchange process. Alternatively, you can specify a DD statement that points to the current GDG(0). If you set APPEND=Y in the TRACING statement, PowerExchange appends messages to the end of the current GDG.

To name the alternative log files, PowerExchange uses the PFX=*prefix* value that you specify in the TRACING statement. The following table describes how the format of the file name varies by operating system and component:

Operating System	Log File Name Format
i5/OS	<p>The PowerExchange Listener uses the following file-naming convention:</p> <pre>datalib/Plistener_port(prefixnnn)</pre> <p>PowerExchange Condense and other PowerExchange jobs use the following file-naming convention:</p> <pre>datalib/JOBjob_number(prefixnnn)</pre> <p>The variables in these names are:</p> <ul style="list-style-type: none"> - <i>datalib</i>. The PowerExchange data library name specified during PowerExchange installation. - <i>job_number</i>. The i5/OS job number for the tracing subtask, DTLTRTSK, that runs under PowerExchange Condense or other PowerExchange jobs. - <i>listener_port</i>. The PowerExchange Listener port number. - <i>prefixnnn</i> is the PFX parameter value with an appended sequential number from 001 through 999.
Linux and UNIX	<p>The PowerExchange Listener uses the following file-naming convention:</p> <pre>logpath/prefix/DTLLST1.plistener_port.nnnn.log</pre> <p>The PowerExchange Logger for Linux, UNIX, and Windows uses the following file-naming convention:</p> <pre>logpath/prefix/PWXCL.tyyyymmddhhmmss.ppid.nnnn.log</pre> <p>Other tasks use the following file-naming convention:</p> <pre>logpath/prefix/module.tyyyymmddhhmmss.ppid.nnnn.log</pre> <p>The variables in these names are:</p> <ul style="list-style-type: none"> - <i>logpath</i>. The value of the LOGPATH statement in the dbmover.cfg file. - <i>prefix</i> is the PFX parameter value. - <i>listener_port</i>. The PowerExchange Listener port number. - <i>module</i>. The name of the PowerExchange module that is running, such as DTLURDMO for the DTLURDMO utility or DTLOBCDRVR for PowerCenter operations. - <i>nnn</i>. A sequential number from 001 through 999. - <i>pid</i>. The process ID of the PowerExchange task. - <i>yyyymmddhhmmss</i>. The time stamp for the point in time when the file was created.

Operating System	Log File Name Format
Windows	<p>The PowerExchange Listener uses the following file-naming convention:</p> <pre>logpath\prefix\DTLLST1.plistener_port.nnnn.log</pre> <p>A PowerExchange Listener Service uses the following file-naming convention:</p> <pre>logpath\prefix\DTLLSTNT.plistener_port.nnnn.log</pre> <p>The PowerExchange Logger for Linux, UNIX, and Windows uses the following file-naming convention:</p> <pre>logpath\prefix\PWXCCL.tyyyymmddhhmmss.ppid.nnnn.log</pre> <p>Other tasks use the following file-naming convention:</p> <pre>logpath\prefix\module.tyyyymmddhhmmss.ppid.nnnn.log</pre> <p>The variables in these names are:</p> <ul style="list-style-type: none"> - <i>logpath</i>. The value of the LOGPATH statement in the dbmover.cfg file. - <i>prefix</i> is the PFX parameter value. - <i>listener_port</i>. The PowerExchange Listener port number. - <i>module</i>. The name of the PowerExchange module that is running, such as DTLURDMO for the DTLURDMO utility or DTLODBCVR for PowerCenter operations. - <i>nnn</i>. A sequential number from 001 through 999. - <i>pid</i>. The process ID of the PowerExchange task. - <i>yyyymmddhhmmss</i>. The time stamp for the point in time when the file was created.
z/OS	<p>The PowerExchange Listener uses the following file-naming convention:</p> <pre>prefix.sysid.Plistener_port.Nnnn</pre> <p>All other PowerExchange batch jobs and started tasks use the following file-naming convention:</p> <pre>prefix.job_name.job_number.sysid.Nnnn</pre> <p>The variables in these names are:</p> <ul style="list-style-type: none"> - <i>prefix</i> is the high-level qualifier or qualifiers that you specify in the PFX parameter. Maximum length of the entire prefix is 16 characters. - <i>sysid</i>. The system ID of the z/OS system on which the batch job or started task runs. - <i>listener_port</i>. The PowerExchange Listener port number. - <i>job_name</i>. The job name of the batch job or started task. - <i>job_number</i>. The JES job number, which begins with JOB for batch jobs and STC for started tasks. - <i>nnn</i>. A generated sequential number from 001 through 999.

Determining the Size of the Alternative Log Data Sets on z/OS

PowerExchange uses DCB attributes to allocate alternative log data sets on z/OS.

On z/OS, PowerExchange uses the following DCB attributes to allocate alternative log data sets:

- Block size (BLKSIZE) is 27998.
- Data set organization (DSORG) is PS.
- Record format (RECFM) is VB.
- Record length (LRECL) is the value of the RECLEN parameter in the TRACING statement.

You can use the following formulas to calculate the primary space and secondary space, in blocks, of an alternative log data set:

```
primary space = log_size / 27998
secondary space = primary_space / 10
```

These formulas include the following variables:

- *log_size* is the value that is specified for the SIZE parameter in the TRACING statement, converted from megabytes to bytes.
- *primary_space* is the *log_size* divided by 27998, where 27998 is the block size in bytes.
- *secondary_space* is the *primary_space* divided by 10.

For example, if you specify 100 for the SIZE parameter, use the following formulas to calculate primary space and secondary space:

```
primary space = 104,857,600 / 27998 = 3745 blocks
secondary space = 3745 / 10 = 374 blocks
```

Allocation Options for Alternative Log Data Sets on z/OS

On z/OS, PowerExchange dynamically allocates alternative log data sets to log messages from PowerExchange jobs and started tasks if you define the TRACING statement in the DBMOVER configuration file. Alternatively, you can specify a DTLLOG DD statement in the JCL for the PowerExchange jobs and started tasks that issue messages.

Use any of the following methods to allocate alternative log data sets on z/OS:

- **Dynamic allocation.** If you define the TRACING statement, PowerExchange dynamically allocates log data sets except for any PowerExchange job or started task for which you add a DTLLOG nn DD statement in the JCL. To name the dynamically allocated log data sets, PowerExchange uses the PFX parameter value in the TRACING statement in conjunction with the file-naming convention for the PowerExchange component and platform. To determine the number of log data sets, PowerExchange uses the FILENUM parameter value in the TRACING statement.
- **Single DTLLOG01 DD statement with SYSTOUT=***. To simplify configuration and make finding message output for a specific job execution easier, Informatica recommends that you define a single DTLLOG01 DD statement that points to SYSTOUT=* in the JCL for PowerExchange jobs and started tasks that issue messages. When you run one of these jobs or tasks, all of its message output is written to a single SYSTOUT data set. In this case, PowerExchange does not use the PFX, APPEND, FILENUM, and VIEW parameters from the TRACING statement. However, if you have other jobs and started tasks that use dynamic allocation, PowerExchange honors the PFX, APPEND, FILENUM, and VIEW parameters for these jobs and tasks.
- **DTLLOG nn DD statements with different data set names.** You can specify DTLLOG nn DD statements with different data set names in the JCL for PowerExchange jobs or started tasks. You must specify the PFX parameter in the TRACING statement, but PowerExchange ignores it. Also, PowerExchange ignores the FILENUM parameter for all jobs that include the DTLLOG nn DD statements. To switch to another alternative log file when the current one becomes full, PowerExchange uses the same round-robin algorithm as it uses for dynamically allocated log data sets.

The variable nn is a number from 01 through 99. You can specify up to 99 DTLLOG nn DD statements.

- **Single DTLLOG01 DD statement that specifies a GDG.** To use a GDG for alternative logging on z/OS, specify GDG(0) and DISP=SHR in a single DTLLOG01 DD statement in the PowerExchange Listener JCL. Also, allocate and create at least one generation of the GDG before starting the PowerExchange Listener. By using GDG(0), you can use APPEND=Y in the TRACING statement to enable PowerExchange to resume logging messages to the current generation. Also, you can use VIEW=Y to view log records in the GDG while the PowerExchange Listener task is active. If you specify APPEND=N, PowerExchange creates a new generation when the PowerExchange Listener starts. PowerExchange does not limit the number of GDG generations. The FILENUM parameter in the TRACING statement is ignored. Instead, the LIMIT value in the GDG definition limits the number of generations. When this limit is reached, the operating system automatically removes the oldest generation from the GDG and deletes it if the SCRATCH parameter is defined for the GDG.

Restriction: You cannot use DFSMS compression for alternative log data sets.

Before using DTLLOGnn DD statements, review the following considerations:

- You still must specify the DTLLOG DD statement for the following purposes:
 - To write initial startup messages
 - To write messages if PowerExchange encounters an error allocating or writing to alternative log data sets

The DTLLOG DD statement can point to a permanent data set or a SYSOUT data set. Informatica recommends that you specify use SYSOUT=* as a safeguard against any failures with alternative log files and to prevent the out-of-space conditions.

- Unless you specify a DTLLOGnn DD statement in the JCL, PowerExchange dynamically allocates alternative log files.
- To prevent loss of PowerExchange log and trace information, use unique data set names for each PowerExchange job with DTLLOGnn DD statements.
- If you use a GDG for alternative logging, add a single DTLLOG01 DD statement that specifies GDG(0) in the JCL.
- PowerExchange jobs and started tasks that write to alternative log files include the PowerExchange Listener, PowerExchange Condense, PowerExchange Agent, and netport jobs. Also, the Adabas, IDMS, and IMS log-based ECCRs and the Datacom table-based ECCR can use alternative log files.

However, the PowerExchange Logger, batch VSAM ECCR, CICS/VSAM ECCR, and IMS synchronous ECCR do not write messages to the PowerExchange message log or alternative log files.

RELATED TOPICS:

- [“TRACING Statement” on page 193](#)

PowerExchange Message Destination Overrides

PowerExchange can write messages to several locations. If you want to redirect specific messages to locations other than the default location, you can define a DTLMSG file that contains message destination override statements.

Messages can be written to the following override destinations:

- PowerExchange log file (DTLLOG or detail.log) or an alternative log file
- Standard output (stdout)

- z/OS or i5/OS system operator console

The ability to redirect messages to destinations other than the default location is useful when you need to enable automation or adhere to installation-specific output handling standards. Usually, message destination overrides are defined for only a small number of messages.

Also, in the DTLMSG file, you can suppress specific messages so that they are not written to any location. Before you do so, consider the implications of suppressing particular messages.

To configure message destination overrides, create a file called DTLMSGO at one of the following locations:

- On z/OS, a sample DTLMSGO member is provided in the RUNLIB library. You can copy and customize this member, or create a sequential data set and define the message destination override statements in this data set. In the JCL for tasks and jobs that produce messages with message destination overrides, add a DTLMSGO DD statement that points to the customized DTLMSGO member or sequential data set.
- On i5/OS, the DTLMSGO member resides in the *dtllib*/DTLMSG file.
- On Linux, UNIX, and Windows, the *dtlmsgo.txt* file resides in the PowerExchange base installation directory. A sample file, called *dtlmsgo_sample.txt*, is provided in that directory.

In the DTLMSGO file, specify message destination override statements. A message override statement includes the message number followed by one or more message destination options. The message number must exist in the DTLMSG file (i5/OS or z/OS) or *dtlmsg.txt* file (Linux, UNIX, or Windows).

Note: On z/OS, PowerExchange can issue messages other than those in the DTLMSG file. The message destination overrides do not apply to these messages. These messages are issued by the following PowerExchange components using the EDMMSG DD statement:

- PowerExchange Agent (a small number of messages are issued from the DTLMSG file)
- PowerExchange Logger
- Log-Read API and Log-Write API
- Environmental Change Capture Routines (ECCRs), including the batch VSAM, CICS/VSAM, DB2 for z/OS, and IMS synchronous ECCRs

RELATED TOPICS:

- [“PowerExchange Message Logs” on page 218](#)

DTLMSGO File and Statements

The DTLMSGO file contains statements that indicate destinations for specific message numbers. Define a DTLMSGO file if you want to redirect certain messages to destinations other than the default location.

Override destinations include the PowerExchange message log file or alternative log files, stdout, or system operator console.

In the DTLMSGO file, you can also suppress a message so that it is not displayed in any of these locations

DTLMSGO Statements - Syntax and Parameters

Use the correct syntax and valid parameter values when defining message destination override statements in the DTLMSGO file.

Syntax:

Use the following syntax:

```
/*Message sent to multiple destinations
msg_number,{msg_dest},{msg_dest}, ...
/*Message destination is not overridden
```



```

msg_number
/*Suppress message
msg_number,NONE

```

The following syntax rules apply:

- Comment lines must begin with "/*" in column 1.
- If the same message appears more than once in the DTLMSGO file, only the last occurrence is used.

Parameters:

msg_number

Required. Specify a message number. The message number must exist in the DTLMSG file on i5/OS or z/OS or in the dtlmsg.txt file on Linux, UNIX, or Windows. Message numbers are 5-digit numbers with no prefix.

Specify the message number with or without the leading zeroes. You cannot specify a mask with wildcards for a *msg_number* value.

msg_dest

Optional. Select one or more options to indicate the override destinations for the specified message number. Options are:

- **null**. If no destination is specified for the message number, the message is not redirected to a different destination. The message is treated as if it is not included in the DTLMSGO file.
- **LOG**. The message is routed to the primary PowerExchange message log or to an alternative log file. If alternative logging is enabled but the alternative logging subtask has not initialized, the message is routed to the PowerExchange message log file.
- **NONE**. The message is suppressed. The message is not displayed in any location. Do not specify this value with any other option.
- **STDOUT**. The message is routed to the standard output location, which varies by system.
 On z/OS, STDOUT is sent to the SYSPRINT DD data set if specified or to a dynamically allocated temporary JES SPOOL data set.
 On i5/OS, STDOUT is sent to QPRINT if PowerExchange is running as a background job or to the terminal if PowerExchange is running interactively.
 On Linux, UNIX, and Windows, STDOUT is sent to the terminal by default unless you redirect it elsewhere.
- **SYSCONS**. On i5/OS and z/OS, the message is displayed in the system operator interface.
 On z/OS, the message is processed by the Write-To-Operator (WTO) macro and displayed in the system operator console. No routing code is associated with the WTO so the ROUTCODE value that is specified in the DEFAULT statement of CONSOLxx is used for routing.
 On i5/OS, the message is sent to the operator message queue.
 On Linux, UNIX, and Windows, SYSCONS is ignored.

Note: The *null* and NONE options are mutually exclusive. The other options can be specified together.

DTLMSGO File Example

The following example DTLMSGO file includes six message destination override statements:

```

/*
/* Sample file to change message destinations
/*
/* region size msg to console
594,SYSCONS
/* use pwx defaults

```

```
00595
/* multiple destinations
00650,SYSCONS,LOG
00651,SYSCONS,LOG,STDOUT
/* suppress stats msg
00408,NONE
/*
607,STDOUT,LOG
/*
```

The message numbers 594 and 607 are entered without leading zeroes. The message number 00595 has no override destination. The message numbers 00650, 00651, and 607 have multiple override destinations.

CHAPTER 5

SMF Statistics Logging and Reporting

This chapter includes the following topics:

- [SMF Statistics Logging and Reporting Overview, 227](#)
- [Configuring Statistics Logging to SMF, 228](#)
- [Configuring Statistics Logging to a PowerExchange File, 229](#)
- [Generating Reports of SMF Statistics, 230](#)
- [Viewing Statistics Written to an SMF Data Set from the PowerExchange Navigator, 235](#)
- [Viewing Statistics Written to a PowerExchange File from the PowerExchange Navigator, 239](#)
- [SMF Record Format, 240](#)

SMF Statistics Logging and Reporting Overview

PowerExchange can use the IBM System Management Facilities (SMF) to log statistics records for monitoring PowerExchange activity and performance.

The PowerExchange Listener produces the following types of SMF statistics records:

- Connection record, driven by the PowerExchange Listener
- Start record, driven by a PowerExchange process, netport job, PowerExchange Listener, or subtask
- End record, for all processes including those that ended abnormally
- Interval statistics record, for all processes

These records report the following information:

- Record header and section descriptor information
- SMF triplet information
- General information, including the PowerExchange version and build, component name, process name and ID, node name, client session information, map name, return codes, CPU time, length of extended section descriptors, and zIIP times
- PowerExchange Listener statistics, including the Listener job or task name, node name, IP address, starting and ending times, port number, number of connections, CPU time, number of bytes and messages sent, and number of bytes and messages received

- PowerExchange client statistics, including the start and end times, CPU time, number of bytes and messages sent, number of bytes and messages received, and return codes
- Access-method statistics, including the access method type, number of rows and bytes read, and number of rows and bytes written
- DB2 for z/OS function and access-method statistics, including the CPU time, number of DB2 statements executed, number of rows returned, DB2 SSID, DB2 plan name, connection type, authorization ID, correlation ID, and number of rows by DML change type

To log SMF statistics records to SMF data sets, you must specify the STATS statement with the SMF parameter and a record type in the DBMOVER configuration file. Consult with your system programmer or capacity planning team to determine the SMF record type to use. The record type can be in the range of 128 through 255.

Alternatively, for testing purposes, you can log statistics records to a sequential data set on z/OS. To log statistics to a sequential data set, specify the STATS statement with the FILE parameter. The FILE parameter must specify a pre-allocated data set.

In both logging scenarios, you can optionally specify an interval at which to log statistics records. The records then show cumulative statistics for all tasks that are active at the time the statistics are generated. You can use these statistics to monitor the activity and progress of long-running tasks and to compare activity over multiple interval-reporting periods. Interval logging does not report the following information:

- Activity by tasks that started and completed between two interval reports
- Activity by tasks that ended since the last interval report

After you log SMF statistics, you can view them from the PowerExchange Navigator. If you logged SMF statistics to SMF archived data sets, you also can generate reports based on the SMF record type. To generate reports, use the DTLSMFRP program that PowerExchange provides.

RELATED TOPICS:

- [“Generating Reports of SMF Statistics” on page 230](#)
- [“Configuring Statistics Logging to a PowerExchange File” on page 229](#)
- [“Configuring Statistics Logging to SMF” on page 228](#)
- [“SMF Record Format” on page 240](#)
- [“Viewing Statistics Written to a PowerExchange File from the PowerExchange Navigator” on page 239](#)
- [“Viewing Statistics Written to an SMF Data Set from the PowerExchange Navigator” on page 235](#)

Configuring Statistics Logging to SMF

PowerExchange does not write statistics records to SMF by default. To enable statistics logging to SMF, specify the STATS statement with the SMF parameter in the DBMOVER configuration member on z/OS.

Use the following syntax:

```
STATS=(SMF,record_type_number,{interval|0})
```

For example:

```
STATS=(SMF,255,0)
```

The following table describes the parameters in the statement:

Parameter	Description
<i>record_type_number</i>	A number that identifies the SMF record type. If you use multiple PowerExchange Listeners, Informatica recommends that you use a unique record-type number for each Listener so that you can collect statistics separately for each one. Important: Ensure that the record-type number is not in use by any other third-party program or product on the system. Valid values are 128 through 255.
<i>interval</i>	The interval, in minutes, at which PowerExchange writes statistics records to SMF. Valid values are 0 and any number from 5 through 120. Default is 0, which disables interval processing. PowerExchange writes an SMF record when the PowerExchange Listener shuts down.

Also, ensure that all libraries in the STEPLIB of the PowerExchange Listener JCL are APF-authorized.

After you configure the STATS statement and APF-authorize the STEPLIB libraries, restart the PowerExchange Listener.

RELATED TOPICS:

- [“Viewing Statistics Written to an SMF Data Set from the PowerExchange Navigator” on page 235](#)

Configuring Statistics Logging to a PowerExchange File

If you need to get SMF statistics quickly, such as for testing purposes, log statistics records to a sequential data set on z/OS instead of to SMF.

To log statistics to a sequential data set, specify the STATS statement with the FILE parameter in the DBMOVER configuration file. Then, restart the PowerExchange Listener. By default, PowerExchange does not collect the statistics.

Use the following syntax for the STATS statement:

```
STATS=(FILE,dataset_name,{interval|0})
```

For example:

```
STATS=(FILE,DTLUSR.V101.PWXFILE,0)
```

The following table describes the parameters in the statement:

Parameter	Description
<i>dataset_name</i>	The fully qualified name of the sequential data set to which to write statistics records. You must have pre-allocated this data set.
<i>interval</i>	The interval in minutes at which to log statistics. Valid values are 0 and any number from 5 through 120. Default is 0, which disables interval processing. PowerExchange writes an SMF record when the PowerExchange Listener shuts down.

Generating Reports of SMF Statistics

You can use the PowerExchange DTLSMFRP program to generate reports of PowerExchange activity based on statistics records that the PowerExchange Listener writes to SMF. The program reads the statistics records from SMF archived data sets only.

To generate reports from SMF statistics records, you must first define a STATS statement in the PowerExchange DBMOVER configuration file. The STATS statement specifies the SMF record type of the records that contain PowerExchange statistics.

The DTLSMFRP program generates the following types of reports, ordered by SMF record:

- **Time and CPU statistics.** Reports the following information, ordered by SMF record:
 - SMF record type
 - Process ID of the PowerCenter client connection
 - Job name and user ID associated with the client
 - Start time of the PowerExchange Listener or client
 - End time of the client connection to the Listener
 - CPU time used during the reporting interval and as of the end of the session
- **PowerCenter session information.** Reports the map name and PowerCenter session ID.
- **Access method statistics.** Reports the access method, the file name or table name that the access request used, the number of rows that were read and that were written, and the number of bytes that were read and that were written.
- **Db2 statistics.** If you have a Db2 for z/OS source, reports the following information ordered by SMF record:
 - Db2 SSID
 - Plan name
 - Connection type (CONNTYPE)
 - Authorization ID (AUTHID)
 - Correlation ID
 - SQL statements processed
 - Number of rows processed
 - SQL return code (SQLCODE)

- Reason code (RSNCODE)
- Number of updates, deletes, and inserts processed
- CPU usage

The DTLSMFRP program can also create dumps of SMF information that Informatica Global Customer Support can use for diagnosing problems, or that third-party reporting tools can use to determine the SMF record layouts that the PowerExchange Listener produces.

Tip: If you want the report to contain information from a single PowerExchange Listener, assign a unique SMF record type to the STATS parameter in each PowerExchange Listener DBMOVER configuration file. Otherwise, the report will contain information from every PowerExchange Listener that has the same SMF record type.

Defining the Set of SMF Data Sets to Use for Reporting

To report SMF statistics that were recorded during a PowerCenter session run, configure the set of SMF archived data sets that contain the SMF records that were written during this period.

Note: This procedure uses the MVS SWITCH SMF command. If you do not have authority to issue this command, contact your SMF or system administrator.

1. In the z/OS console, switch the SMF data set to which the PowerExchange Listener writes SMF records by submitting the following command:

```
/SWITCH SMF
```

This step creates a base archived data set.

2. Record the generation data set name or generation data group (GDG) number of the base data set that was just archived.

The next generation data set (+1 data set) will be used as the starting data set for reporting.

3. Run the PowerCenter session with the PowerExchange Listener that produces SMF statistics records.
4. Switch the SMF data set again to create another archived SMF data set.
5. Determine the generation data set name or GDG number of the last archived SMF data set.
6. In the SMFFILE DD statement of the DTLSMFRP JCL, concatenate all of the archived SMF data sets starting from the +1 generation data set up to the last archived data set. These archived data sets include the PowerExchange SMF records that were produced when the session run. The DTLSMFRP program will use these data sets to generate reports that contain the SMF statistics for PowerExchange and the PowerCenter session.

Note: Many SMF archived data sets can be created during a session run.

For more information, see ["Sample JCL for Generating Reports" on page 231](#).

Sample JCL for Generating Reports

To generate SMF statistics reports, you must define JCL statements that run the DTLSMFRP utility with the correct PARMs and input SMF archived data sets.

Use the following sample JCL, which is also available in the PowerExchange RUNLIB library as member DTLSMFRP:

```
//SMF EXEC PGM=DTLSMFRP,PARM='0,128,SYSOUT=A' <-- 'Report number, SMF record type, SYSOUT
Class'
//STEPLIB DD DSN=your.PWX.load,DISP=SHR
//SYSOUT DD SYSOUT=*
//SYSPRINT DD SYSOUT=*
//SYSUDUMP DD SYSOUT=*
/*
```

```
//SMFFILE DD DISP=SHR,DSN=your.SMF.GnnnnV00 <-- Starting SMF generation data set
// DD DISP=SHR,DSN= DSN=your.SMF.GnnnnV00 <-- +1 Next SMF generation data set
// ... <-- +2 to +(x-1) generation data sets
// DD DISP=SHR,DSN= DSN=your.SMF.GnnnnV00 <-- Ending SMF data set
```

In the EXEC statement, specify the reporting program name of DTLSMFRP, which this job step will execute. The PARM option specifies the report number and SMF record type.

- The *report number* can be one of the following values:
 - **0.** Generates an SMF statistics report. The SMF record content determines the report content.
 - **1.** Generates a formatted dump of each SMF block in the input data set.
 - **2.** Generates a formatted dump of each SMF record in the input data set.
 - **3.** Generates a formatted dump of each SMF section for records of the specified record type.
 - **4.** Generates a list of all SMF data elements found in each SMF record of the specified record type.

Use option 0 to generate reports.

Options 1 through 4 are primarily for diagnostic use by Informatica technical staff. Option 4 can also be used by third-party SMF reporting vendors to determine the SMF record layouts that the PowerExchange Listener produces.

- The *SMF record type* is the numeric identifier for the record type, as specified in a STATS statement in the DBMOVER member. This value can be in the range of 128 through 255.

Important: The STATS statement in the DBMOVER configuration file for the PowerExchange Listener specifies the SMF record type that is written by the PowerExchange Listener. To create reports on behalf of the PowerExchange Listener, the DTLSMFRP report parameter SMF record type must match the SMF record type identified in the PowerExchange Listener STATS statement.

The optional SYSOUT parameter specifies where to route the reports. For example, to route the reports to SYSOUT class A, specify:

```
//SMF EXEC PGM=DTLSMFRP,PARM='0,252,SYSOUT=A'
```

The default class is X.

In the SMFFILE DD statement, provide a concatenated list of all SMF archived data sets that contain the PowerExchange-written SMF records to use for reporting. These archived data sets are typically generation data sets in a generation data group (GDG).

Example Reports

Use the example reports to determine which reports best suit your requirements and to understand the information that each report presents.

Note: Each report shows statistics by record. The **Record** column lists the relative record numbers of the records in the input SMF archived data sets. You can use these record numbers to link information in different reports. If a record number is not listed in a report, the record contains no information for the report. If a record number is repeated in a report, such as the Access Method Statistics report, multiple instances of a SMF section might occur in the SMF record.

Example 1. Time and CPU Statistics

Example report:

Time and CPU Statistics							Page: 1	
Record Type	Proc_Id	Job_Name	User_Id	Start_Time	End_Time	CPU_Time_1	CPU_Time_2	Version
2126	PWX	Intvl		2018 03/28 17:33:08		0.018992	0.471135	10.2.0
2127	ClI	Intvl	73309	A02DTLSN xxINFAXx	2018 03/28 17:33:25	0.031312	5.225357	10.2.0
2128	ClI	Intvl	73313	A02DTLSN ausqa	2018 03/28 17:37:25	2.861607	5.504249	10.2.0
2129	ClI	Intvl	73315	A02DTLSN AUSQA	2018 03/28 17:37:44	21.006851	28.564011	10.2.0
2289	ClI	End	73318	A02DTLSN AUSQA	2018 03/28 17:39:19	0.056402	0.056402	10.2.0
2561	PWX	Intvl		2018 03/28 17:33:08	2018 03/28 17:39:33	0.022712	0.493847	10.2.0

2562	Cli	Intvl	73309	A02DTLSN	xxINFAXx	2018 03/28 17:33:25		0.039725	5.265082	10.2.0
2563	Cli	Intvl	73313	A02DTLSN	ausqa	2018 03/28 17:37:25		0.432298	5.936546	10.2.0
2564	Cli	Intvl	73319	A02DTLSN	AUSQA	2018 03/28 17:39:30		0.031786	0.031786	10.2.0
2565	Cli	Intvl	73315	A02DTLSN	AUSQA	2018 03/28 17:37:44		7.693156	36.257172	10.2.0
2696	Cli	End	73320	A02DTLSN	AUSQA	2018 03/28 17:40:26	2018 03/28 17:40:26	0.027886	0.027886	10.2.0
2999	Cli	End	73321	A02DTLSN	AUSQA	2018 03/28 17:41:05	2018 03/28 17:41:05	0.028508	0.028508	10.2.0
3025	PWX	Intvl				2018 03/28 17:33:08		0.022894	0.516741	10.2.0
3026	Cli	Intvl	73309	A02DTLSN	xxINFAXx	2018 03/28 17:33:25		0.031705	5.296787	10.2.0
3027	Cli	Intvl	73313	A02DTLSN	ausqa	2018 03/28 17:37:25			5.936546	10.2.0
3028	Cli	Intvl	73319	A02DTLSN	AUSQA	2018 03/28 17:39:30			0.031786	10.2.0
3029	Cli	Intvl	73315	A02DTLSN	AUSQA	2018 03/28 17:37:44		2.280185	38.537354	10.2.0

Report fields:

- **Record.** The relative record number of the SMF record within the input SMF archived data sets.
- **Type.** The type of SMF record. Valid values are:
 - **PWX Intvl.** The PowerExchange interval record that contains the amount of CPU time that the PowerExchange Listener address space used, up to the end of the reporting interval.
 - **Cli Intvl.** The client interval record that contains the amount of CPU time that an active PowerCenter client session used, up to the end of the reporting interval.
 - **Cli End.** The client end record that contains the total amount of CPU time that the PowerCenter client session used, up to the end of the session.
- **Proc_ID.** The unique process identifier that is assigned to a client connection.
- **Job_Name.** The job name that is associated with the client.
- **User Id.** The user ID that the client uses.
- **Start_Time.** The start time of the PowerExchange Listener or client, depending on the Type of record.
- **End_Time.** The ending time of the client connection to the PowerExchange Listener.
- **CPU_Time_1.** The CPU time that was used up to the end of the current reporting interval. This value is reported if interval-based reporting is enabled in the STATS statement of the PowerExchange DBMOVER configuration file.
- **CPU_Time_2.** The total CPU time that was used up to the end of the client session.

Note: The CPU_TIME_1 value will match the CPU_TIME_2 value when the client session ends or the PowerExchange Listener connection ends.

Example 2. PowerCenter Session Information

Example report:

Record	Map_Name	Session_Id	PWC Information	Page: 1
1332			i_mhvaix0264_ausqa/wf_2ST_lim011_lim012_dsnc/s_2st_lim011_lim012_dsna	
1468	lim012		i_mhvaix0264_ausqa/wf_2ST_lim011_lim012_dsnc/s_2st_lim011_lim012_dsna	
1469			i_mhvaix0264_ausqa/wf_2ST_lim011_lim012_dsnc/s_2st_lim011_lim012_dsna	
1617			i_mhvaix0264_ausqa/wf_2ST_lim011_lim012_dsnc/s_2st_lim011_lim012_dsna	
1708			i_mhvaix0264_ausqa/wf_2ST_lim011_lim012_dsna/s_2st_lim011_lim012_dsna	
1837	lim012		i_mhvaix0264_ausqa/wf_2ST_lim011_lim012_dsnc/s_2st_lim011_lim012_dsna	
1846	lim012		i_mhvaix0264_ausqa/wf_2ST_lim011_lim012_dsnc/s_2st_lim011_lim012_dsna	
1847	lim012		i_mhvaix0264_ausqa/wf_2ST_lim011_lim012_dsna/s_2st_lim011_lim012_dsna	
1848			i_mhvaix0264_ausqa/wfaccounts_ic/s_accounts_ic_to_accounts	
1849			i_mhvaix0264_ausqa/wf_2ST_lim011_lim012_dsna/s_2st_lim011_lim012_dsna	
1925			i_mhvaix0264_ausqa/wfaccounts_ic/s_accounts_ic_to_accounts	
1998			i_mhvaix0264_ausqa/wfaccounts_fic/s_accounts_fic_to_accounts	
2128	lim012		i_mhvaix0264_ausqa/wf_2ST_lim011_lim012_dsna/s_2st_lim011_lim012_dsna	
2129			i_mhvaix0264_ausqa/wf_2ST_lim011_lim012_dsna/s_2st_lim011_lim012_dsna	
2289			i_mhvaix0264_ausqa/wfqaalltypes_fic/s_qaalltypes_fic_to_qaalltypes2	
2563	lim012		i_mhvaix0264_ausqa/wf_2ST_lim011_lim012_dsna/s_2st_lim011_lim012_dsna	

Report fields:

- **Record.** The relative record number of the SMF record within the input SMF archived data sets.
- **Map_Name.** The map name that the PowerCenter client uses. This value can be a data map or extraction map name.
- **Session_Id.** The PowerCenter session ID.

Example 3. Access Method Statistics

Example report:

			Access Method Information		Page: 1	
Record	Access	File/Type Table	Read	Rows Written	Read	Bytes Written
2129	DB2	CONNECTION	0	0	0	0
2129	DB2	ADA8421.LIM11_TGT	0	0	0	0
2129	DB2	ADA8421.LIM11_TGT	0	4853	0	140427
2129	DB2	ADA8421.LIM11_TGT	0	0	0	0
2129	DB2	ADA8421.LIM12_TGT	0	0	0	0
2129	DB2	ADA8421.LIM12_TGT	0	4921	0	141351
2129	DB2	ADA8421.LIM12_TGT	0	0	0	0
2129	DB2	ADA8421.LIM12_TGT	0	7918	0	225448
2129	DB2	ADA8421.LIM11_TGT	0	7798	0	222586
2289	DB2	CONNECTION	0	0	0	0
2289	DB2		0	0	0	0
2563	CAPXRT	CONNECTION	6075	0	0	0
2563	CAPXRT	DTLTEMP.UOWCTRL DTL UOW TABLE	0	0	0	0
2563	CAPXRT	didsna.lim011_LIM11	0	0	0	0
2563	CAPXRT	didsna.lim012_LIM12	0	0	0	0
2564	DB2	CONNECTION	0	0	0	0
2564	DB2		0	0	0	0
2565	DB2	CONNECTION	0	0	0	0
2565	DB2	ADA8421.LIM11_TGT	0	0	0	0
2565	DB2	ADA8421.LIM11_TGT	0	1773	0	51414
2565	DB2	ADA8421.LIM11_TGT	0	0	0	0
2565	DB2	ADA8421.LIM12_TGT	0	0	0	0
2565	DB2	ADA8421.LIM12_TGT	0	1749	0	49958

Report fields:

- **Record.** The relative record number of the SMF record within the input SMF archived data sets.
- **Access.** The access method that the client session used to access the PowerExchange source.
- **File/Type or Table.** The file name or table name that the access request used.
- **Rows Read.** The number of rows that were read.
- **Rows Written.** The number of rows that were written.
- **Bytes Read.** The number of bytes that were read.
- **Bytes Written.** The number of bytes that were written.

Example 4. Db2 for z/OS Statistics

Example report:

Record	SSID	Plan	Conn_Type	Auth_Id	Corr_Id	Stmts	Rows	SQLCODE	RSNCODE	Update	Delete	Insert	CPU
3771	DSNA	AUBK1020	CAF	AUSQA	PWX73315	1	24341	0	0	0	0	0	9.206862
3771	DSNA	AUBK1020	CAF	AUSQA	PWX73315	1	0	0	0	32869	0	0	13.745912
3771	DSNA	AUBK1020	CAF	AUSQA	PWX73315	1	0	0	0	0	0	0	0.003656
3771	DSNA	AUBK1020	CAF	AUSQA	PWX73315	1	24198	0	0	0	0	0	9.068873
3771	DSNA	AUBK1020	CAF	AUSQA	PWX73315	1	0	0	0	32949	0	0	13.629420
3771	DSNA	AUBK1020	CAF	AUSQA	PWX73315	1	0	0	0	0	39355	0	9.477407
3771	DSNA	AUBK1020	CAF	AUSQA	PWX73315	1	0	0	0	0	39170	0	9.568233
4062	DSNA	AUBK1020	CAF	AUSQA	PWX73315	1	0	0	0	0	0	0	0.005295
4062	DSNA	AUBK1020	CAF	AUSQA	PWX73315	1	29817	0	0	0	0	0	11.241923
4062	DSNA	AUBK1020	CAF	AUSQA	PWX73315	1	0	0	0	40177	0	0	16.763763
4062	DSNA	AUBK1020	CAF	AUSQA	PWX73315	1	0	0	0	0	0	0	0.003656
4062	DSNA	AUBK1020	CAF	AUSQA	PWX73315	1	29635	0	0	0	0	0	11.065903
4062	DSNA	AUBK1020	CAF	AUSQA	PWX73315	1	0	0	0	40189	0	0	16.585022
4062	DSNA	AUBK1020	CAF	AUSQA	PWX73315	1	0	0	0	0	46561	0	11.353901
4062	DSNA	AUBK1020	CAF	AUSQA	PWX73315	1	0	0	0	0	46576	0	11.511835

Report fields:

- **Record.** The relative record number of the SMF record within the input SMF archived data sets.
- **SSID.** The Db2 subsystem ID.
- **Plan.** The Db2 plan name that is used to access the Db2 for z/OS source.
- **Conn_Type.** The type of Db2 attachment facility that the client session used to connect to Db2. Valid values are:
 - **CAF.** Call attachment facility.
 - **RRSAF.** Resource Recovery Services attachment facility.
- **Auth_Id.** The Db2 authorization ID that PowerExchange used for the client session.

- **Corr_Id.** The Db2 correlation ID that PowerExchange used for the client session.
- **Stmts.** The number of SQL statements that the session processed.
- **Rows.** The number of rows that Db2 returned.
- **SQLCODE.** The Db2 code.
- **RSNCODE.** The Db2 reason code for the specified SQLCODE.
- **Update.** The number of updates that were processed.
- **Delete.** The number of deletes that were processed.
- **Insert.** The number of inserts that were processed.
- **CPU.** The amount of Db2 CPU time that was charged to the client request.

Viewing Statistics Written to an SMF Data Set from the PowerExchange Navigator

To view SMF statistics that are stored in an SMF data set on z/OS from the PowerExchange Navigator, you must use the sample pwxstat.file data map that has the same release level as the current PowerExchange Listener. Otherwise, inconsistent results might occur.

The pwxstat.file.dmp data map file is located in the *PowerExchange_installation*\examples\datamaps directory on the PowerExchange Navigator system. Do not edit this file.

Before you can view SMF statistics in the PowerExchange Navigator, you must extract the SMF data from the system SMF data set and place the data into an intermediate file for viewing.

1. On the z/OS system, use the SMF SWITCH console command to refresh the collection of SMF statistics.
2. Run the IBM IFASMFDP utility to extract the SMF records that pertain to PowerExchange. For example, use the following utility SYSIN parameters:

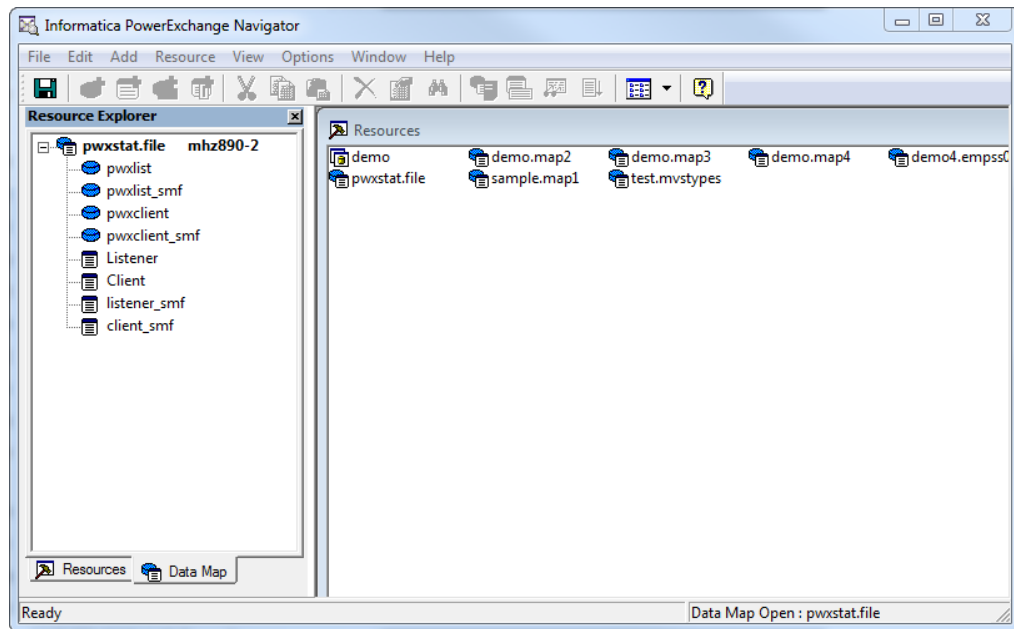
```
SID(xxxx)           <-Where xxxx is the LPAR from which to extract data
INDD(SYSUT1,OPTIONS(DUMP))
OUTDD(SYSUT2,TYPE(255)) <-Where 255 is the SMF record number in the PWX STATS statement
```

3. Stop the PowerExchange Listener on the z/OS system if the Listener is running.
4. Add the STATS statement with the SMF parameter to the DBMOVER configuration file on the z/OS system. For example:

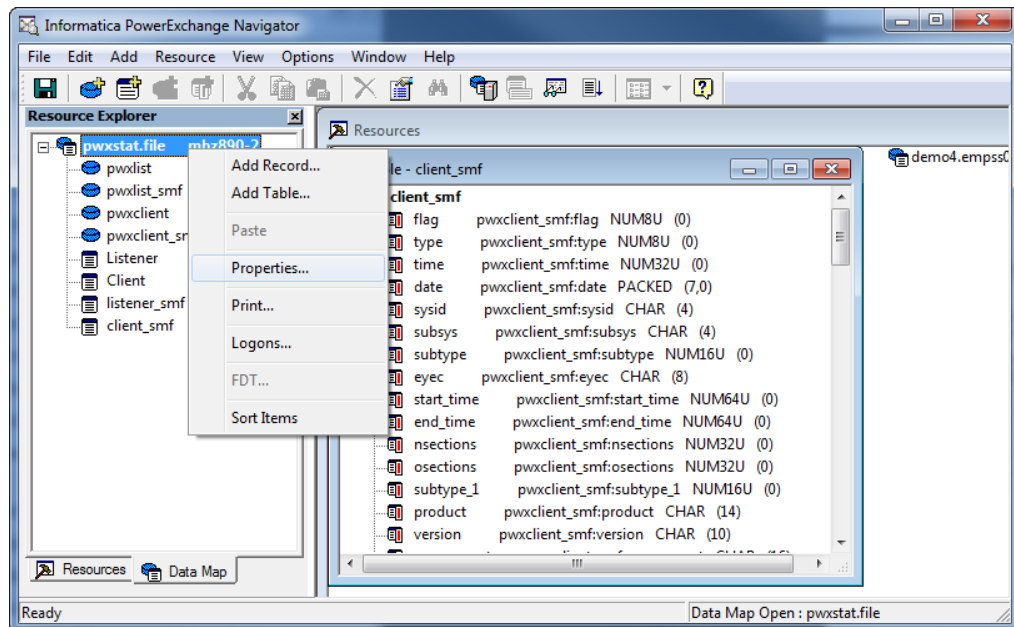
```
STATS=(SMF,255,0)
```

5. Start or restart the PowerExchange Listener.

6. In the PowerExchange Navigator, expand the **Data Maps** node in the Resource Explorer and open the pwxstat.file data map.



7. On the **Data Map** tab in the Resource Explorer, right-click the pwxstat.file and click **Properties**.

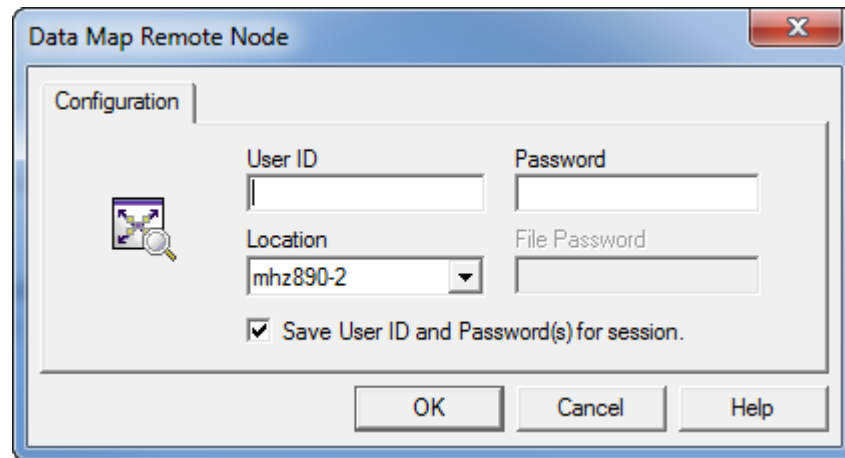


8. In the **Data Map Properties** dialog box, click the **SEQ Access Method** tab.

The screenshot shows the 'Data Map Properties' dialog box with the 'SEQ Access Method' tab selected. The 'File Name' field contains 'stqa.smf.dataset'. The 'Record Format' section has 'Variable' selected with 'VBI2' in the dropdown. Other options like 'Fixed', 'Default', 'Field Separator', 'Merge Adjacent Separators', 'Field Delimiter', 'Encoding', and 'Codepage' are also visible. At the bottom, there are 'Skip First' and 'Records from File' fields, a 'View File...' button, and a 'File List Processing' checkbox. The 'Data Map' icon is on the left, and 'OK', 'Cancel', and 'Help' buttons are at the bottom right.

9. In the **File Name** field, enter the fully qualified name of the sequential data set to which you extracted SMF data in step 2. Then click **OK**.
10. To generate SMF statistics that are available for viewing in the PowerExchange Navigator, stop and restart the PowerExchange Listener on z/OS.
11. In the PowerExchange Navigator, run a database row test on any table with the suffix "_SMF" in the pwxstat.file data map.
- Select the table on the **Data Map** tab, and then select **File > Database Row Test**.
 - If the data map has not been previously saved to the z/OS system, the PowerExchange Navigator displays a message prompt that asks if you want to send the data map to the remote location. Click

Yes. Then in the **Data Map Remote Node** dialog box, enter connection information for the remote z/OS location and click **OK**.

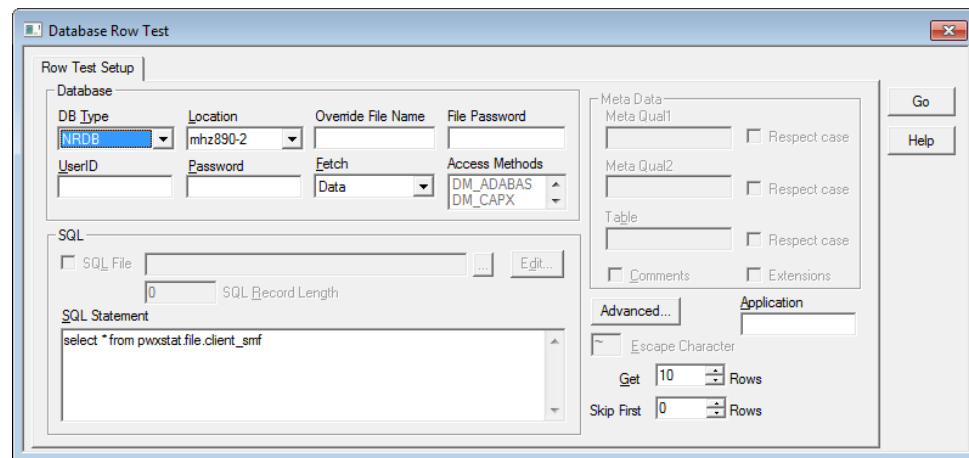


The **Data Map Remote Node** dialog box is shown with the **Configuration** tab selected. It contains the following fields and options:

- User ID**: Text input field.
- Password**: Text input field.
- Location**: Dropdown menu with **mhz890-2** selected.
- File Password**: Text input field.
- ☒ **Save User ID and Password(s) for session.**

Buttons at the bottom: **OK**, **Cancel**, **Help**.

- c. In the **Database Row Test** dialog box, ensure that the following settings are specified:
- In the **DB Type** list, select **NRDB**.
 - In the **Location** list, select the z/OS node name.
 - In the **Fetch** list, select **Data**.



The **Database Row Test** dialog box is shown with the **Row Test Setup** tab selected. It contains the following sections and settings:

- Database** section:
 - DB Type**: **NRDB** (selected)
 - Location**: **mhz890-2** (selected)
 - Override File Name**: Empty text field
 - File Password**: Empty text field
 - User ID**: Empty text field
 - Password**: Empty text field
 - Fetch**: **Data** (selected)
 - Access Methods**: **DM_ADABAS**, **DM_CAPX**
- SQL** section:
 - ☐ **SQL File**: Empty text field with **Edit...** button
 - SQL Record Length**: **0**
 - SQL Statement**: `select * from pwxstat.file_client_smf`
- Meta Data** section:
 - Meta Qual1**: Empty text field, ☐ **Respect case**
 - Meta Qual2**: Empty text field, ☐ **Respect case**
 - Table**: Empty text field, ☐ **Respect case**
 - ☐ **Comments**, ☐ **Extensions**
- Advanced...** button
- Application** section:
 - ☐ **Escape Character**
 - Get**: **10** Rows
 - Skip First**: **0** Rows

Buttons on the right: **Go**, **Help**.

- d. Click **Go**.

The **Database Row Test Output** window displays the PowerExchange statistics that were collected from the SMF data set.

RELATED TOPICS:

- [“Configuring Statistics Logging to SMF” on page 228](#)

Viewing Statistics Written to a PowerExchange File from the PowerExchange Navigator

To view SMF statistics that were written to a PowerExchange sequential data set on z/OS from the PowerExchange Navigator, you must use the sample pwxstat.file data map that has the same release level as the current PowerExchange Listener. Otherwise, inconsistent results might occur.

The pwxstat.file.dmp data map file is located in the *PowerExchange_installation\examples\datamaps* directory on the PowerExchange Navigator system. Do not edit this file.

Warning: If you write statistics directly to a sequential data set or GDG, PowerExchange operational problems might occur.

1. On the z/OS system, stop the PowerExchange Listener.
2. In the DBMOVER configuration file on the z/OS system, add a STATS statement that includes the FILE parameter. In the FILE parameter, specify the fully qualified name of the sequential data set to which to write statistics. For example:

```
STATS=(FILE,DTLUSR.V800.PWXFILE,0)
```

3. Start the PowerExchange Listener again.
4. In the PowerExchange Navigator, expand the **Data Maps** node in the Resource Explorer and open the pwxstat.file data map.
5. On the **Data Map** tab in the **Resource Explorer**, right-click the pwxstat.file and click **Properties**.
6. In the **Data Map Properties** dialog box, click the **SEQ Access Method** tab.
7. In the **File Name** field, enter the name of the sequential data set that you specified in the STATS statement in the DBMOVER configuration file in step 2. Then click **OK**.
8. To generate PowerExchange statistics that will be available for viewing from the PowerExchange Navigator, stop and restart the PowerExchange Listener on z/OS.
9. In the PowerExchange Navigator, run a database row test on any table in the pwxstat.file data map that does NOT have the "_SMF" suffix.
 - a. Select the table on the **Data Map** tab, and then click **File > Database Row Test**.
 - b. If the data map has not been previously saved to the z/OS system, the PowerExchange Navigator displays a message prompt that asks if you want to send the data map to the remote location. Click **Yes**. Then in the **Data Map Remote Node** dialog box, enter connection information for the remote z/OS location and click **OK**.
 - c. In the **Database Row Test** dialog box, ensure that the following settings are specified:
 - In the **DB Type** list, select **NRDB**.
 - In the **Location** list, select the z/OS node name.
 - In the **Fetch** list, select **Data**.
 - d. Click **Go**.

The **Database Row Test Output** window displays the SMF statistics that were written to the PowerExchange file.

SMF Record Format

When you log statistics to SMF, each statistics record contains a standard SMF header, including the subtype identifier.

When you log statistics to a file, each statistics record includes an SMF header for consistency. However, the detail in this record is limited.

Both formats include a PowerExchange header with triplets that define occurrences, lengths, and offsets for all record sections.

The following table shows the type of content that PowerExchange logs for each record subtype:

Sub Type	General Section	Extended Section	Function/Access Method Section
1	X	-	-
2	X	-	X
3	X	X	X
4	X	X	X
13	X	X	X
14	X	X	X

Standard SMF Header with Subtype

The following table describes the fields in the standard SMF header with subtype section:

Field	Decimal Offset	Hexadecimal Offset	Length	Format	Description
SMFxLEN	0	00	2	Binary	Record length. The SMFxLEN and SMFxSEG fields, which total 4 bytes, form the record descriptor word (RDW). The first 2 bytes (SMFxLEN) contain the logical record length.
SMFxSEG	2	02	2	Binary	Segment descriptor, which is used for variable block spanned records. If the record is not spanned, these 2 bytes are set to hexadecimal zeros.
SMFxFLG	4	04	1	Binary	Header flag byte.
SMFxRTY	5	05	1	Binary	Record type. Hexadecimal values are from 0 through FF.
SMFxTME	6	06	4	Binary	The amount of time since midnight, in hundredths of a second, that the record was moved into the SMF buffer. In record types 2 and 3, this field indicates the time that the record was moved into the dump data set.

Field	Decimal Offset	Hexadecimal Offset	Length	Format	Description
SMFxDTE	10	0A	4	Packed	Date when the record was moved into the SMF buffer, in the form 00 yyddd F or 0 cyyddd F. Where: <ul style="list-style-type: none"> - c is 0 for 19xx, and 1 for 20xx. - yy is the current year from 0 through 99. - ddd is the current day from 1 through 366. - F is the sign. In record types 2 and 3, this field indicates the date that the record was moved to the dump data set.
SMFxsID	14	0E	4	EBCDIC	The system ID taken from the SID parameter.
SMFxsSI	18	12	4	EBCDIC	Subsystem identification. This field is a 4-byte character value set by the SUBSYS=option that is specified in the SMF macros.
SMFxsTY	22	16	2	Binary	Record subtype. Specifies the PowerExchange record description type. This value is the same as the value in the PWXGSTP field in the General section.

For more information about the standard SMF header with record subtypes, see the IBM guide, *z/OS V1R7.0 MVS System Management Facilities (SMF)*.

SMF Triplet Section Descriptor

The following table describes the fields in the SMF triplet section:

Field	Decimal Offset	Hexadecimal Offset	Length	Format	Description
SMFxNS	0	0	2	Binary	Number of triplets.
SMFxSOF	0	4	4	Binary	Offset to section, relative to the start of the SMF header.
SMFxLNS	0	0	2	Binary	Length of section.
SMFxNOS	0	0	2	Binary	Number of sections.

PowerExchange Header/Descriptor

The following table describes the fields in the PowerExchange header/descriptor section:

Field	Decimal Offset	Hexadecimal Offset	Length	Format	Description
PWXHEYE	0	0	8	Char	Eye catcher. Value is "PWXHDR_".
PWXHSTC	8	8	8	Binary	Starting clock time.
PWXHENC	16	10	8	Binary	Ending clock time.
PWXHESN	24	18	4	Binary	Number of extended sections.
PWXHOSN	28	1C	4	Binary	Offset to the first extended section, relative to this section.

Extended Section Descriptor

The following table describes the fields in the extended section descriptor:

Field	Decimal Offset	Hexadecimal Offset	Length	Format	Description
PWXETYP	0	0	4	Binary	1=PowerExchange Listener section 2=Exception section. Reserved for future use. 3=File section 4=DB2 section 5=Client section
PWXESCT	4	4	4	Binary	Count of sections of the same type that follow.
PWXESLN	8	8	4	Binary	Length of sections that follow.

General Section

The following table describes the fields in the general section:

Field	Decimal Offset	Hexadecimal Offset	Length	Format	Description
PWXGSTP	0	0	2	Binary	Numeric value that represents the subtype. Values: x'0001' = Connection request x'0002' = Start x'0003' = Interval x'0004' = End x'0013' = PowerExchange Listener interval x'0014' = PowerExchange Listener end (same as in the standard SMF header)
PWXGPRD	2	2	14	Char	PowerExchange product. Contains the literal "PowerExchange".
PWXGVER	16	10	10	Char	The PowerExchange version and build.
PWXGCMP	26	1A	16	Char	A component name, such as "Listener" or a utility name.
PWXGJOB	42	2A	8	Char	PowerExchange Listener or the name of the process.
PWXGNDN	50	32	8	Char	Node name, when applicable.
PWXGPID	58	3A	5	Char	Task or process ID, which is a value from 1 through 99999.
Reserved	63	3F	3	Binary	Reserved for future use.
PWXGCID	66	42	8	Char	Session ID of the client. The value is taken from the SESSID parameter of the connection parameters. For ODBC, the DTLSESSID SQL escape sequence.
PWXGCIP	74	4A	4	Binary	IP address of the client.
PWXGUID	78	4E	128	Char	User ID for the client.
PWXGSTT	206	CE	20	Timestamp	Start timestamp for this client session.
Reserved	226	E2	8	-	Reserved for future use.
PWXGETT	234	EA	20	Timestamp	Ending timestamp for this client session.
Reserved	254	FE	2	-	Reserved for future use.
PWXGRC	256	100	8	Char	Return code.

Field	Decimal Offset	Hexadecimal Offset	Length	Format	Description
PWXGRSN	264	108	8	Char	Reason code.
PWXGADI	272	110	24	Char	Additional information that is used to describe the return code or reason code.
PWXGCIP	296	128	16	Char	Client IP string.
PWXGSEQ	312	138	4	Binary	Sequence number.
Reserved	316	13C	8	-	Reserved for future use.
PWXGCPU	324	144	8	Char	CPU used.
PWXGLES	332	14C	4	Binary	Length of extended section descriptors.
PWXGSID	336	150	256	Char	Name of the PowerCenter Session ID. Only present for PowerCenter connections.
PWXGMNM	592	250	11	Char	Map name.
PWXGTCP	603	25B	8	Binary	zIIP ¹ qualified time.
PWXGTOT	611	263	8	Binary	Time spent on the zIIP ¹ .
PWXGTOF	619	26B	8	Binary	zIIP ¹ time offloaded to the Central Processor.
1. System z Integrated Information Processor (zIIP) times are cumulative. PowerExchange uses only one Workload Manager (WLM) enclave.					

Extended Section for the PowerExchange Listener

The following table describes the fields in the extended section for the PowerExchange Listener:

Field	Decimal Offset	Hexadecimal Offset	Length	Format	Description
PWXLJNM	0	0	8	Char	Name of the job or started task.
PWXLNDN	8	8	8	Char	Node name.
PWXLIP	16	10	16	Char	IP address.
PWXLSTT	32	20	20	Timestamp	Start timestamp. For DB2, the format is: YYYYMMDDHHMMSSNNNNNN
PWXLENT	52	34	20	Timestamp	Ending timestamp. For DB2, the format is: YYYYMMDDHHMMSSNNNNNN

Field	Decimal Offset	Hexadecimal Offset	Length	Format	Description
PWXLSSK	72	48	8	Binary	Start time (STCK).
PWXLESK	80	50	8	Binary	Ending time (STCK).
PWXLPR	88	58	4	Binary	Port number.
PWXLNSC	92	5C	4	Binary	Number of connections.
Reserved	96	60	4	Binary	Reserved for future use.
PWXLACN	100	64	4	Binary	Number of accepted connections.
PWXLNCN	104	68	4	Binary	Number of connections refused because the memory was exceeded.
PWXLNCR	108	6C	4	Binary	Number of connections refused because of the task limit.
PWXLCPU	112	70	8	Binary	CPU time of task, in microseconds.
PWXLBSN	120	78	4	Binary	Number of bytes sent.
PWXLMSN	124	7C	4	Binary	Number of messages sent.
PWXLBRV	128	80	4	Binary	Number of bytes received.
PWXLMRV	132	84	4	Binary	Number of messages received.

Extended Section for the PowerExchange Client

The following table describes the fields in the extended section for the PowerExchange client:

Field	Decimal Offset	Hexadecimal Offset	Length	Format	Description
PWXCSSK	0	0	8	Binary	Start time (STCK).
PWXCESK	8	8	8	Binary	Ending time (STCK).
PWXCPCU	16	10	8	Binary	CPU time of task, in microseconds.
PWXCBSN	24	18	4	Binary	Number of bytes sent.
PWXCMSN	28	1C	4	Binary	Number of messages sent.
PWXCBRV	32	20	4	Binary	Number of bytes received.
PWXCMRV	36	24	4	Binary	Number of messages received.
PWXCRC1	40	28	4	Binary	Return code.

Field	Decimal Offset	Hexadecimal Offset	Length	Format	Description
PWXCRC2	44	2C	4	Binary	Return code.
PWXCRC3	48	30	4	Binary	Return code.

Extended Section for Data Access

The following table describes the fields in the extended section for data access:

Field	Decimal Offset	Hexadecimal Offset	Length	Format	Description
PWXDAM	0	0	1	Char	Access method type.
Reserved	1	1	7	Binary	Reserved for future use.
PWXDRR	18	12	4	Binary	Number of rows read.
PWXDBR	22	16	8	Binary	Number of bytes read.
PWXDRW	30	1E	4	Binary	Number of rows written.
PWXDBW	34	22	8	Binary	Number of bytes written.
PWXSAMN	42	2A	20	Char	The name of the access method.

Function/Access Method Section (DB2)

The following table describes the fields in the function/access method section for DB2:

Field	Decimal Offset	Decimal Offset	Length	Format	Description
PWX2CPU	0	0	8	Binary	CPU time, in microseconds.
PWX2NST	8	8	4	Binary	The number of DB2 statements executed.
PWX2NR	12	C	4	Binary	The number of rows returned.
PWX2SSID	16	10	4	Char	The DB2 SSID.
PWX2PLAN	20	14	8	Char	The DB2 plan name.
PWX2CNTY	28	1C	8	Char	The connection type. Values are "RRSAF" and "CAF."
PWX2AUTH	36	24	8	Char	The authorization ID.

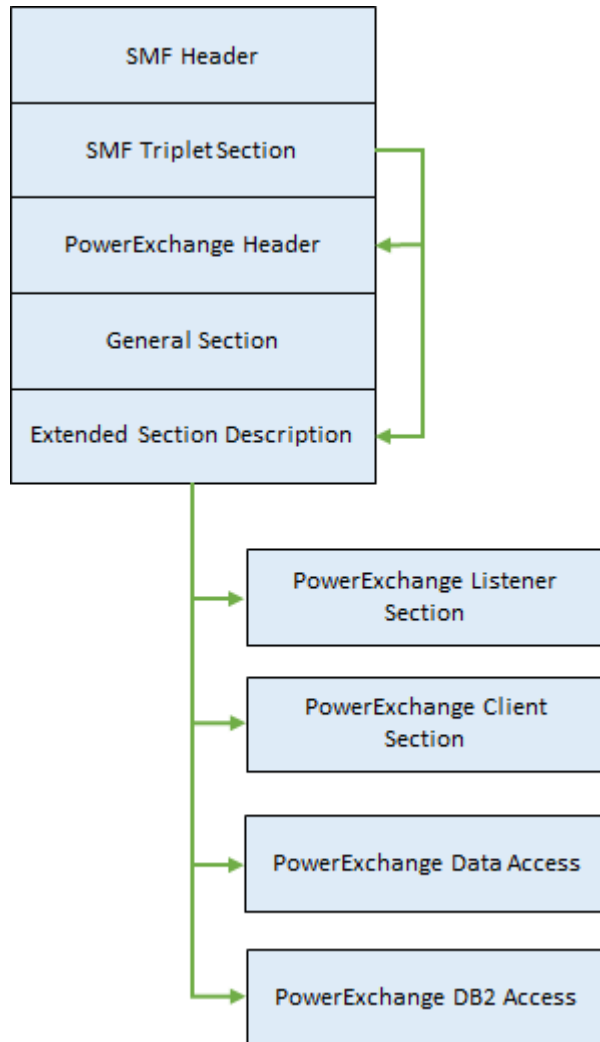
Field	Decimal Offset	Decimal Offset	Length	Format	Description
PWX2CORR	44	2C	32	Char	Correlation ID, in the following format: <i>PWX nnnnn</i> Where <i>nnnnn</i> is a unique number that the PowerExchange Listener generates. PowerExchange generates a correlation ID for each subtask that the PowerExchange Listener generates.
PWX2SRC	76	4C	4	Binary	The SQL code.
PWX2SRS	80	50	4	Binary	The reason code.
PWX2RUC	84	54	4	Binary	The number of rows updated.
PWX2RDL	88	58	4	Binary	The number of rows deleted.
PWX2RIN	92	5C	4	Binary	The number of rows inserted.
PWX2QTY	96	60	1	Binary	DB2 connection type similar to DWHCATYP. Values are: X'02' = CAF X'12' = RRSF

SMF Record Layout

If you write statistics to the SMF data set, the record layout consists of header and section components.

The SMF triplet section points to the PowerExchange header and Extended section description. The Extended section description contains several types of sections.

The following image shows the SMF record layout:

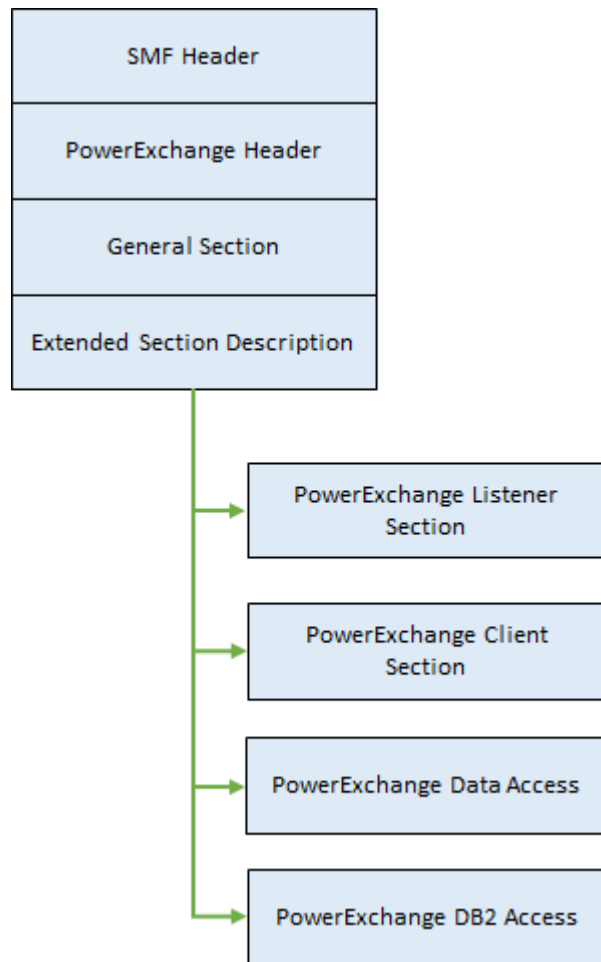


PowerExchange File Record Layout

If you write statistics to a PowerExchange sequential data set that is specified in the STATS FILE parameter, the record layout consists of header and section components.

The Extended section description contains several types of sections.

The following image shows the record layout:



CHAPTER 6

PowerExchange Security

This chapter includes the following topics:

- [PowerExchange Security Overview, 250](#)
- [SECURITY Statement, 251](#)
- [i5/OS Security, 255](#)
- [Linux, UNIX, and Windows Security, 260](#)
- [z/OS Security, 263](#)
- [PowerExchange Passphrases, 279](#)
- [PowerExchange LDAP User Authentication, 280](#)
- [PowerExchange Selective Sign-on, 293](#)
- [PowerExchange Sign-on File, 294](#)

PowerExchange Security Overview

You can use PowerExchange security options to authenticate users for connection to a PowerExchange Listener and to authorize user access to resources that are required by PowerExchange jobs and tasks.

PowerExchange security options are available for all operating systems. Some security options are specific to an operating system or data source type.

The SECURITY statement in the DBMOVER configuration file determines the level of security that PowerExchange provides.

Optionally, on all operating systems, PowerExchange can use its selective sign-on capability to authorize user connection to PowerExchange. When a PowerExchange Listener accepts a connection over TCP/IP, PowerExchange checks the sign-on file to verify access for the user ID and optionally the IP address.

On i5/OS, PowerExchange provides the following security options:

- PowerExchange can use operating system facilities to authenticate user IDs and passwords for connection to PowerExchange. Also, PowerExchange can use the specified user IDs to check authority to access resources that PowerExchange jobs and tasks need to use.
- If a PowerExchange process is the target of a pwxcmd command, PowerExchange can require a valid operating system user ID and password on the command. PowerExchange can use operating system facilities on the target system to authenticate user IDs and passwords for use of the pwxcmd program.
- PowerExchange can use security objects to control access to LISTTASK and STOPTASK commands issued through the iSeries SNDLSTCMD interface, the PowerExchange Navigator, or the DTLUTSK utility to a PowerExchange Listener running on i5/OS.

- PowerExchange can use security objects to control access to pwxcmd commands issued to a PowerExchange process running on i5/OS.

Note: PowerExchange does not use the sign-on file to control access to pwxcmd commands issued to a PowerExchange process running on i5/OS.

On Linux, UNIX, and Windows, PowerExchange provides the following security options:

- If a PowerExchange process is the target of a pwxcmd command, PowerExchange can require a valid operating system user ID and password on the command. PowerExchange uses operating system facilities on the target system to authenticate user IDs and passwords for use of the pwxcmd program.
- If a PowerExchange application service in the Informatica domain is the target of an infacmd pwx command, PowerExchange can require a valid operating system user ID and password on the command. PowerExchange uses operating system facilities on the target system to authenticate user IDs and passwords for use of the infacmd pwx program. For more information about application services, see the *Informatica Administrator Guide*. For more information about infacmd pwx commands, see the *Informatica Command Reference*.
- PowerExchange can use AUTHGROUP and USER statements in the sign-on file to control access to infacmd pwx commands and pwxcmd commands. You send infacmd pwx commands to a PowerExchange application service and pwxcmd commands to a PowerExchange process that is not managed by an application service.
- On Windows, PowerExchange can use the AUTHGROUP and USER statements in the sign-on file to authorize use of PowerExchange Listener LISTTASK and STOPTASK commands issued through the PowerExchange Navigator.

On z/OS, PowerExchange provides the following security options:

- PowerExchange can use operating system facilities to authenticate user IDs and passwords for connection to PowerExchange. Also, PowerExchange can use the supplied user IDs in conjunction with a z/OS security product such as RACF or ACF2 to check authority to access resources that PowerExchange jobs and tasks need. PowerExchange provides source-specific security options for Adabas, Datacom, DB2, and IMS.
- If a PowerExchange process is the target of a pwxcmd command, PowerExchange can require a valid operating system user ID and password on the command. PowerExchange uses operating system facilities on the target system to authenticate user IDs and passwords for use of the pwxcmd program.
- PowerExchange can use resource profiles to control access to LISTTASK and STOPTASK commands that are issued through the PowerExchange Navigator or the DTLUTSK utility to a PowerExchange Listener running on z/OS.
- PowerExchange can use resource profiles to control access to pwxcmd commands issued to a PowerExchange process running on a z/OS system.

Note: PowerExchange does not use the sign-on file to control access to pwxcmd commands issued to a PowerExchange process running on z/OS.

SECURITY Statement

The SECURITY statement controls PowerExchange user authentication and access to resources and commands.

Use the SECURITY statement in the DBMOVER configuration file to configure the following types of security:

- User authentication to access PowerExchange

- Access to files and data sets by PowerExchange jobs and tasks on z/OS and i5/OS
- User authorization to issue infacmd pwx commands to a PowerExchange application service in the Informatica domain
- User authorization to issue pwxcmd commands to a PowerExchange process
- User authorization to issue PowerExchange Listener LISTTASK and STOPTASK commands from the PowerExchange Navigator

Operating Systems: All

Related Statements: DM_RESOURCE, MVSDDB2AF, and RACF_CLASS

Required: No

Syntax:

```
SECURITY=( {0|1|2}
           , {N|Y}
           [, LDAP]
           [, {ORACLE_LDAP|OPEN_LDAP}]
        )
```

Parameters: The first positional parameter has the following valid values:

{0|1|2}

Controls whether PowerExchange requires users to enter a valid operating system user ID and a password or passphrase. Also controls whether PowerExchange checks user-entered credentials to control access to file and database resources and the issuance of certain PowerExchange commands.

Enter one of the following options:

- **0.** PowerExchange does not require users to specify a valid operating system user ID and password and ignores any credentials that users supply.
On z/OS and i5/OS, PowerExchange uses the user ID under which the PowerExchange Listener or PowerExchange Condense task runs to control access to file resources. PowerExchange passes this user ID to the database system.
On Linux, UNIX, and Windows, PowerExchange uses the user ID under which the PowerExchange Listener task runs to control access to file resources. RDBMS security controls PowerExchange access to database resources based on the user ID that users specify on the PWX connection or in the PowerExchange Logger CAPTURE_NODE_UID parameter.
On all operating systems, PowerExchange does not check user authorization to issue commands. Any user can issue a command.
- **1.** On z/OS and i5/OS, PowerExchange requires users to specify a valid operating system user ID and a password or valid PowerExchange passphrase. PowerExchange checks these credentials when a PowerExchange task starts. Thereafter, PowerExchange controls access to file resources in the same manner as for option 0. For file access, PowerExchange uses the user ID under which the PowerExchange Listener or PowerExchange Condense task runs and passes this user ID to the database system.
On Linux, UNIX, and Windows, unless you specify LDAP for the third parameter of the SECURITY statement on supported systems, PowerExchange does not require users to specify a valid operating system user ID and password to access file or database resources and does not check for these credentials. As for option 0, PowerExchange uses the user ID under which the PowerExchange Listener task runs to control access to file resources. RDBMS security controls PowerExchange access to database resources based on the user ID that users specify on the PWX connection or in the PowerExchange Logger CAPTURE_NODE_UID parameter.

On all operating systems, PowerExchange does not check user authorization to issue commands. Any user can issue a command.

- **2. Provides the most specific level of security.**

- On z/OS, Informatica recommends that you use option 2. PowerExchange controls access based on 1) an MVS user ID and a password or valid PowerExchange passphrase and 2) the access control features of your z/OS security product, such as RACF or ACF2.

To read change data from the change stream, the ECCR must use a valid z/OS user ID and password or passphrase. The PowerExchange Listener checks these credentials when the ECCR task or job starts. To access the database to read data, PowerExchange passes the z/OS user ID and password or passphrase to the database system for database-specific security checking. In conjunction with the z/OS security product and MVS System Authorization Facility (SAF), PowerExchange checks the z/OS user ID and password or passphrase against the CAPX.REG.* resource profiles to control access to capture registrations.

To extract change data, run PowerCenter CDC sessions with a PWXPC connection that specifies a valid z/OS user ID and password or passphrase. For the session to access extraction maps, these user credentials must have READ access to the PowerExchange data set that is defined in the DTLCAMAP DD statement of the PowerExchange Listener JCL.

Note: A connection to DB2 for z/OS through the Call Attachment Facility (CAF) runs under the user ID of the PowerExchange Listener regardless of the security settings. DB2 uses the user ID that is specified on the connection only if the connection type is Recoverable Resource Manager Service Attachment Facility (RRSAF) or if offload processing is enabled.

PowerExchange also uses resource profiles to control who can run the following types of commands:

- pwxcmd commands for a PowerExchange Listener or PowerExchange Condense process that are issued from a Linux, UNIX, or Windows system
- PowerExchange Listener LISTTASK and STOPTASK commands that are issued from the PowerExchange Navigator or the DTLUTSK utility
- On i5/OS, PowerExchange requires users to specify a valid operating system user ID and password or passphrase. PowerExchange checks these credentials when a PowerExchange task starts. PowerExchange Listener subtask processes run under the supplied user ID and password or passphrase. PowerExchange uses this user ID and password or passphrase to control access to PowerExchange files. PowerExchange also passes this user ID and password or passphrase to the database system for data access.

PowerExchange uses security objects to control who can run the following types of commands:

- pwxcmd commands for a PowerExchange Listener or PowerExchange Condense process that are issued from a Linux, UNIX, or Windows system
- PowerExchange Listener LISTTASK and STOPTASK commands that are issued from the SNDLSTCMD interface, the PowerExchange Navigator, or the DTLUTSK utility
- On Linux, UNIX, and Windows, unless you specify LDAP for the third parameter of the SECURITY statement on supported systems, PowerExchange does not require users to specify an operating system ID and password to access PowerExchange files or a database. PowerExchange uses the user ID and password under which the PowerExchange Listener runs or that PowerExchange Logger for Linux, UNIX, and Windows uses to control access to PowerExchange files. RDBMS security controls access to the database.

However, you must specify a valid operating system user ID and password to run the following types of commands:

- An `infacmd pwx` command to a PowerExchange application service in the Informatica domain
- A `pwxcmd` command to a PowerExchange process

PowerExchange checks these user credentials against the USER and AUTHGROUP COMMANDS statements in the sign-on file to determine if a user is authorized to issue an `infacmd pwx` or `pwxcmd` command. In this case, the second positional parameter in the SECURITY statement is ignored.

Default is 0.

The second positional parameter has the following valid values:

{N|Y}

Controls use of PowerExchange selective sign-on file to authorize users to connect to the PowerExchange Listener.

Enter one of the following options:

- **N.** PowerExchange does not use the selective sign-on file.
- **Y.** PowerExchange uses the USER statement with the ALLOW and IP subparameters in the selective sign-on file to restrict users who can connect to the PowerExchange Listener.

Note: If you specify Y and also set the first parameter in the SECURITY statement to 1, PowerExchange uses the TASKCNTRL parameter in the USER statements in the sign-on file to control access to PowerExchange Listener LISTTASK and STOPTASK commands that are issued from the PowerExchange Navigator.

Default is N.

The optional third positional parameter has the following valid value:

LDAP

If you specify LDAP for the third positional parameter and specify 1 or 2 as the first positional parameter, PowerExchange uses LDAP authentication on supported Linux, UNIX, and Windows systems.

If you do not include the third parameter, PowerExchange does not use LDAP authentication.

The fourth positional parameter has the following valid values:

{ORACLE_LDAP|OPEN_LDAP}

If you specify LDAP for the third positional parameter, specifies which set of LDAP client libraries to load.

Enter one of the following options:

- **ORACLE_LDAP.** PowerExchange loads the Oracle LDAP client libraries.
Select this option only if you have an Oracle LDAP installation. PowerExchange does not provide the Oracle LDAP client libraries.
- **OPEN_LDAP.** PowerExchange loads the OpenLDAP client libraries.

Default is ORACLE_LDAP.

Usage Notes:

- In the z/OS Installation Assistant, if you click **Advanced Parms** on the **General Parameters** page, you can define the SECURITY_LEVEL and SECURITY_PWX parameters. The SECURITY_LEVEL parameter corresponds to the first parameter in the SECURITY statement. The SECURITY_PWX parameter corresponds to the second parameter in the SECURITY statement.
- On z/OS, when you set the first parameter of the SECURITY statement to 1 or 2, you must APF-authorize the STEPLIB for the PowerExchange Listener and netport jobs. Otherwise, PowerExchange cannot complete user authentication or control resource access, and instead operates as if you set this parameter to 0.
- If you offload column-level processing for a z/OS data source to the Linux, UNIX, or Windows system where the PowerCenter Integration Service runs, PowerCenter CDC sessions use the **Map Location User** and **Map Location Password** values that you specify on the connection to control access to all resources. The connection must be a PWX NRDB CDC application connection or PWX DB2zOS CDC application connection for which offload processing is enabled.
- If you log data from z/OS data sources to remote PowerExchange Logger for Linux, UNIX, and Windows log files, set the SECURITY option to 2 in the DBMOVER configuration member on z/OS. Ensure that the user ID and password in the PowerExchange Logger for Linux, UNIX, and windows configuration file, pwxocl, is a valid z/OS user ID and password that can pass z/OS security checking. To read captured data from the PowerExchange Logger for z/OS log files, these user credentials must have READ access to CAPX.REG.* resources profiles in the FACILITY class, which are managed by your z/OS security product. Also, for CDC sessions to extract data from the log files, the PWXPC connection must specify the z/OS user ID and password in the **Map Location User** and **Map Location Password** connection attributes. These user credential needs READ access to the CAPX.CND.* resource profiles.

RELATED TOPICS:

- [“i5/OS Security” on page 255](#)
- [“Linux, UNIX, and Windows Security” on page 260](#)
- [“z/OS Security” on page 263](#)
- [“PowerExchange Selective Sign-on” on page 293](#)

i5/OS Security

Use the SECURITY statement in the DBMOVER configuration file on i5/OS to configure PowerExchange security.

You can configure the following types of PowerExchange security on i5/OS:

- **User authentication.** If you specify 1 or 2 in the first parameter of the SECURITY statement, PowerExchange uses a valid operating system user ID and password to authenticate users to connect to and use PowerExchange. Instead of a password, you can specify a valid PowerExchange passphrase for i5/OS. For information about passphrases, see [“PowerExchange Passphrases” on page 279](#). If you also configure PowerExchange selective sign-on, PowerExchange checks operating system user IDs and passwords or passphrases after successful selective sign-on checking.
- **Resource access.** If you specify 2 in the first parameter of SECURITY statement, PowerExchange uses connection user IDs to authorize access to any resources that PowerExchange uses. Otherwise, resource access is controlled by the user ID under which PowerExchange jobs and tasks execute.

- **PowerExchange Listener commands.** If you specify 2 in the first parameter of SECURITY statement, PowerExchange uses security objects to control user access to PowerExchange Listener LISTTASK and STOPTASK commands issued through the iSeries SNDLSTCMD interface, the PowerExchange Navigator, or the DTLUTSK utility. Otherwise, PowerExchange does not control commands issued through the iSeries SNDLSTCMD interface, the PowerExchange Navigator, or the DTLUTSK utility.
- **User authentication for the pwxcmd program.** If you specify 1 or 2 in the first parameter of the SECURITY statement, PowerExchange uses operating system facilities to authenticate users of the pwxcmd program. If you also configure PowerExchange selective sign-on, PowerExchange checks operating system user IDs and passwords after successful selective sign-on checking.
- **Authorization for running pwxcmd commands.** If you specify 2 in the first parameter of the SECURITY statement on the machine that is the target of a command, PowerExchange checks security objects in the PowerExchange data library to determine whether the user ID supplied on the pwxcmd program is authorized to run commands. Otherwise, authority to run pwxcmd commands is not checked.
- **Selective sign-on.** If you specify Y in the second parameter of the SECURITY statement, PowerExchange uses the selective sign-on file to limit the users that connect to PowerExchange. Otherwise, any operating system user ID can connect to PowerExchange.

RELATED TOPICS:

- [“Resource Access Requirements on i5/OS” on page 256](#)
- [“User Authentication for the pwxcmd or infacmd pwx Programs on Linux, UNIX, and Windows” on page 262](#)
- [“Authorization to Run pwxcmd Commands on i5/OS” on page 259](#)
- [“PowerExchange Selective Sign-on” on page 293](#)

Resource Access Requirements on i5/OS

On an i5/OS system, when you specify 2 for the first parameter on the SECURITY statement, PowerExchange uses the connection user ID to control access to resources.

The user ID must have access to specific PowerExchange resources and to any resources that PowerExchange uses.

To use PowerExchange, users require the following authority levels:

- To write messages to the PowerExchange log file, users must have *CHANGE authority to the *datalib* library. By default, PowerExchange create the *datalib* library with a default public authority of *USE.
- To read data maps, users must have *USE authority to the data map library. By default, this library is called STDATAMAPS. To specify a different name, set the DMX_DIR parameter in the DBMOVER member of the CFG file.
- To add, delete, or change data maps, users must have *CHANGE authority.
- To read and write to other libraries, files, and DB2 tables through PowerExchange, users must have the appropriate authority.

Access Requirements for PowerExchange Jobs

To perform bulk data movement and CDC operations, PowerExchange jobs and tasks require certain authority.

The following table lists the access authority that the PowerExchange Listener requires for bulk data processing:

Library and File Name	Authority Required
<i>datalib</i> /CFG	*USE
<i>datalib</i> /LOG	*CHANGE
<i>datalib</i> / <i>Pnnnnn</i> Note: If you use PowerExchange alternative logging, the PowerExchange Listener creates a file named <i>Pnnnnn</i> in the data library, where <i>nnnnn</i> is the port number of the PowerExchange Listener.	*CHANGE
<i>dtllib</i> /DTLMSG	*USE
<i>dtllib</i> /LICENSE	*USE
<i>stdatamaps</i>	*CHANGE

The following table lists the access authority that the PowerExchange Listener and PowerExchange Condense require for bulk data and CDC processing:

File Name	PowerExchange Listener Authority	PowerExchange Condense Authority
<i>cndlib</i> /CFGCOND	*USE	*USE
<i>cndlib</i> /PWXRJNLCKP	*CHANGE	*CHANGE
<i>cpplib</i>	*CHANGE	n/a
<i>datalib</i> /CCT	*CHANGE	*USE
<i>datalib</i> /CDCT	*USE	*CHANGE
<i>datalib</i> /CDEP	*CHANGE	*CHANGE
<i>datalib</i> /CFG	*USE	*USE
<i>datalib</i> /LOG	*CHANGE	*CHANGE
<i>datalib</i> / <i>Pnnnnn</i> Note: If you use PowerExchange alternative logging, the PowerExchange Listener creates a file named <i>Pnnnnn</i> in the data library, where <i>nnnnn</i> is the port number of the PowerExchange Listener.	*CHANGE	n/a
<i>datalib</i> / <i>JOBnnnnn</i> Note: If you use PowerExchange alternative logging, the PowerExchange Condense job creates a file named <i>JOBnnnnn</i> in the data library, where <i>nnnnn</i> is the job number of the PowerExchange Condense job.	n/a	*CHANGE
<i>dtllib</i> /DTLMSG	*USE	*USE

File Name	PowerExchange Listener Authority	PowerExchange Condense Authority
<i>dtllib</i> /LICENSE	*USE	*USE
<i>stdatamaps</i>	*CHANGE	n/a

PowerExchange dynamically creates other objects, such as capture registrations and data maps. For users other than the user ID under which the PowerExchange Listener and PowerExchange Condense run, the default authority to access these objects depends on the following authority settings:

- The Create Authority system value, QCRTAUT, which determines the system-wide public authority for new objects
- The default public authority for the library that contains the objects

Access Requirements for Journals and Files

The user ID under which the PowerExchange Listener and PowerExchange Condense run must have the appropriate authority level to access journals and files that PowerExchange needs to use.

The following table lists the authority requirements by object type:

Object	Authority
Journal	*OBJEXIST
Journal library	*EXECUTE
Journal receivers	*USE
Library that contains journal receivers	*EXECUTE
Files	*USE
Library that contains files	*EXECUTE

Authorization to Run PowerExchange Listener Commands on i5/OS

You can configure PowerExchange to authorize users to issue the PowerExchange Listener LISTTASK and STOPTASK commands through the iSeries SNDLSTCMD interface, the PowerExchange Navigator, and the DTLUTSK utility on i5/OS.

In the DBMOVER configuration file on the i5/OS system that is the target of PowerExchange Listener commands, set the first parameter in the SECURITY statement to 2. PowerExchange checks security objects in the PowerExchange data library to determine whether the user ID supplied on the LISTTASK or STOPTASK command is authorized to run LISTTASK and STOPTASK commands.

During the installation process, PowerExchange creates the security objects for the LISTTASK and STOPTASK commands with a default public authority of *EXCLUDE. To authorize a user to issue LISTTASK and STOPTASK commands, grant the user *USE authority to the security objects for the commands. For example, to authorize a user to issue a LISTTASK command to a PowerExchange Listener running on an i5/OS system, grant the user *USE access to the *datilib*/AUTHTSKLST security object on that system.

The following table lists the access authority that the PowerExchange Listener requires to run LISTTASK and STOPTASK commands:

PowerExchange Listener Command	Library and File Name	Authority Required
LISTTASK	<i>datalib</i> /AUTHSKLST	*USE
STOPTASK	<i>datalib</i> /AUTHSKSTP	*USE

User Authentication for the pwxcmd Program on i5/OS

You can configure PowerExchange to authenticate operating system user IDs and passwords to control the use of the pwxcmd program.

To verify user credentials for pwxcmd program use, complete the following configuration tasks for each PowerExchange installation that is the target of a command from the pwxcmd program:

- In the DBMOVE configuration file on the system that is the target of the command, set the first parameter of the SECURITY statement to 1 or 2.
- Configure the necessary permissions to authenticate operating system user credentials.

Authorization to Run pwxcmd Commands on i5/OS

In addition to authenticating user credentials for pwxcmd program use, you can configure PowerExchange to authorize users to run specific pwxcmd commands.

On the system that is the target of pwxcmd commands, set the first parameter of the SECURITY statement to 2. Then, create security objects, as follows:

- If you completed a first-time installation, PowerExchange creates security objects automatically.
- If you migrated from a previous release, run the following upgrade command to create security objects:

```
CALL PGM(dtllib/CRTDTLENVA) PARM('datalib')
```

Where:

- *dtllib* is the name of the PowerExchange software library that you entered at installation.
- *datalib* is the name for the PowerExchange data library that you entered at installation.

To authorize a user to issue a specific pwxcmd command, set the first parameter of the SECURITY statement to 2 on the system that is the target of pwxcmd commands. Then, grant the user access to the security object for the command. For example, to authorize a user to issue a pwxcmd closeforce command to a PowerExchange Listener running on an i5/OS system, grant the user access to the *datalib*/LCLOSEFRCE security object on that system.

The following table lists the pwxcmd PowerExchange Listener commands and corresponding security objects:

pwxcmd Command	Security Object
close	LCLOSE
closeforce	LCLOSEFRCE

pwxcmd Command	Security Object
listtask	LLISTTASK
stoptask	LSTOPTASK

The following table lists the pwxcmd PowerExchange Condense commands and corresponding security objects:

pwxcmd Command	Security Object
condense	CCONDENSE
displaystatus	CDSPSTATUS
fileswitch	CFILSWITCH
shutcond	CSHUTCOND
shutdown	CSHUTDOWN

When a user issues a pwxcmd command, PowerExchange checks security objects in the PowerExchange data library to determine whether the user ID supplied on the pwxcmd program is authorized to run commands.

Note: To authenticate users to run pwxcmd commands, configure PowerExchange to check user credentials for the pwxcmd program.

RELATED TOPICS:

- [“User Authentication for the pwxcmd Program on i5/OS” on page 259](#)
- [“PowerExchange Sign-on File” on page 294](#)

Linux, UNIX, and Windows Security

Use the SECURITY statement in the dbmover.cfg configuration file on Linux, UNIX, and Windows to configure PowerExchange security.

On Linux, UNIX, and Windows, you can configure the following types of PowerExchange security:

- **PowerExchange Listener commands.** On Windows, if you set the first parameter in the SECURITY statement to 1 and the second parameter to Y, PowerExchange uses the TASKCNTRL parameter in USER statements in the sign-on file to control access to PowerExchange Listener LISTTASK and STOPTASK commands issued through the PowerExchange Navigator. Otherwise, PowerExchange does not control access to commands issued through the PowerExchange Navigator.
- **User authentication for the pwxcmd and infacmd pwx programs.** If you set the first parameter in the SECURITY statement to 1 or 2, PowerExchange uses operating system facilities to authenticate users of the pwxcmd and infacmd pwx programs.

If you set the second parameter to Y to configure PowerExchange selective sign-on, PowerExchange checks operating system user IDs and passwords after successful selective sign-on checking.

- **Authorization for running pwxcmd and infacmd pwx commands.** If you set the first parameter in the SECURITY statement to 2 and set the second parameter to Y, PowerExchange checks these user credentials against the USER and AUTHGROUP COMMANDS statements in the sign-on file the PowerExchange sign-on file to check user authority to run specific pwxcmd and infacmd pwx commands. Otherwise, authority to run these commands is not checked.
- **Selective sign-on.** If you set the second parameter in the SECURITY statement to Y, PowerExchange uses the selective sign-on file to limit the users that connect to PowerExchange and check user authority to issue PowerExchange Listener commands from the PowerExchange Navigator. Otherwise, any user ID can connect to PowerExchange and issue PowerExchange Listener commands from the PowerExchange Navigator on Windows.
- **LDAP Authentication.** If you set the first positional parameter in the SECURITY statement to 1 or 2 and the third positional parameter to LDAP, PowerExchange uses LDAP authentication to verify the credentials of a client that is trying to connect to PowerExchange on a Linux, UNIX, or Windows system.

RELATED TOPICS:

- [“Authorization to Run pwxcmd and infacmd pwx Commands on Linux, UNIX, and Windows” on page 261](#)
- [“PowerExchange Selective Sign-on” on page 293](#)
- [“PowerExchange LDAP User Authentication” on page 280](#)

Authorization to Run PowerExchange Listener Commands on Windows

You can configure PowerExchange to authorize users to issue the PowerExchange Listener LISTTASK and STOPTASK commands through the the PowerExchange Navigator on Windows.

In the dbmover.cfg configuration file on the PowerExchange Navigator system, set the first parameter in the SECURITY statement to 1 and the second parameter to Y. PowerExchange checks the TASKCNTRL parameter in USER statements in the sign-on file to determine whether the user is authorized to run LISTTASK and STOPTASK commands through the PowerExchange Navigator. For more information, see [“PowerExchange Sign-on File” on page 294](#).

Authorization to Run pwxcmd and infacmd pwx Commands on Linux, UNIX, and Windows

In addition to authenticating user credentials for pwxcmd and infacmd pwx program use, you can configure PowerExchange to authorize users to run specific pwxcmd and infacmd pwx commands.

On the system that is the target of pwxcmd or infacmd pwx commands, set the first parameter of the SECURITY statement to 2. Then, configure a sign-on file on each PowerExchange installation that is the target of pwxcmd or infacmd pwx commands.

In the PowerExchange sign-on file, you can create authorization groups, which PowerExchange uses in conjunction with user definitions to authorize users to run one or more pwxcmd or infacmd pwx commands. If you do not create specific authorization groups to authorize users to issue commands, you must include the following statements in the sign-on file on all target systems to which users issue commands:

```
AUTHGROUP= (ANY, COMMAND= ( * ) )
USER= ( * , ALLOW=Y, AUTHGROUP=ANY )
```

When a user issues a command, PowerExchange checks the PowerExchange sign-on file to determine whether the user ID supplied on the command is authorized to run the command.

Note: To authenticate users to run pwxcmd and infacmd pwx commands, configure PowerExchange to check user credentials for the pwxcmd and infacmd pwx programs.

RELATED TOPICS:

- [“User Authentication for the pwxcmd Program on z/OS” on page 277](#)
- [“PowerExchange Sign-on File” on page 294](#)

User Authentication for the pwxcmd or infacmd pwx Programs on Linux, UNIX, and Windows

You can configure PowerExchange to authenticate operating system user IDs and passwords to control the use of the pwxcmd or infacmd pwx programs.

To verify user credentials for pwxcmd or infacmd pwx program use, complete the following configuration tasks for each PowerExchange installation that is the target of pwxcmd or infacmd pwx commands:

- In the dbmover.cfg file on the system that is the target of the command, set the first parameter in the SECURITY statement to 1 or 2.
- Configure the necessary permissions to authenticate operating system user credentials.

User Authentication on Linux and UNIX

For PowerExchange to authenticate user credentials on Linux and UNIX systems, the operating system must use shadowed passwords.

PowerExchange reads user IDs and passwords from the /etc/passwd and /etc/shadow file. By using the crypt function, PowerExchange verifies that the passwords supplied on pwxcmd or infacmd pwx commands matches the password for the user ID.

Restriction: PowerExchange authentication of user credentials does not support Linux and UNIX systems with security that is not backwardly compatible with the shadow password file and crypt function.

1. Change the owner of pwxauth.exe to root.
2. Change the group of pwxauth.exe to the group of the PowerExchange Listener.
3. Set the setuid bit on pwxauth.exe.

PowerExchange provides a sample script called setup_pwxauth in the PowerExchange installation directory that makes the necessary changes to pwxauth.exe. To run this script, issue the following command:

```
setup_pwxauth listener_primary_group
```

For the variable *listener_primary_group*, specify the group name of the PowerExchange Listener. The PowerExchange Listener should be the only UID in the specified group.

User Authentication on Windows

To verify user credentials on Windows systems, PowerExchange uses the LogonUser API.

On Windows 2000 and previous releases, the PowerExchange Listener user ID must have the SE_TCB_NAME privilege. In User Manager, this privilege is the “Act as part of the Operating System” right. On Windows Server 2003 and later releases and on Windows XP and later releases, no additional Windows configuration is necessary.

For more information about the LogonUser API, see the Microsoft article, “How to validate user credentials on Microsoft operating systems” at <http://support.microsoft.com/kb/180548>.

Restriction: When you enable security for the pwxcmd and infacmd pwx programs on Windows, PowerExchange does not accept guest as an authorized user account.

z/OS Security

To configure z/OS security, define a SECURITY statement in the DBMOVER configuration member in conjunction with other security methods, such as operating system facilities, resource profiles, and the selective sign-on file.

You can configure the following types of PowerExchange security:

Note: On z/OS, when you set the first parameter in the SECURITY statement to 1 or 2, you must APF-authorize the STEPLIB for the PowerExchange Listener and netport jobs. Otherwise, PowerExchange cannot complete user authentication or control resource access, and instead operates as if you set the first parameter in the SECURITY statement to 0.

- **User authentication.** Set the first parameter in the SECURITY statement to 1 or 2. PowerExchange uses a valid MVS user ID and password to authenticate users to connect to and use PowerExchange. Instead of a password, you can specify a valid PowerExchange passphrase for z/OS. For information about passphrases, see [“PowerExchange Passphrases” on page 279](#). If you also configure PowerExchange selective sign-on, PowerExchange checks operating system user IDs and passwords or passphrases after successful selective sign-on checking.
- **PowerCenter CDC session access.** Set the first parameter in the SECURITY statement to 2 to enable PowerCenter CDC sessions to use the z/OS user ID and password that is specified on the PWXPC connection to extract data. The connection user ID and password must have READ access to the data set defined in the DTLCAMAP DD statement of the PowerExchange Listener JCL.

Note: A connection to DB2 for z/OS through the Call Attachment Facility (CAF) runs under the user ID of the PowerExchange Listener regardless of the security settings. DB2 uses the user ID supplied on the connection only if the connection type is Recoverable Resource Manager Service Attachment Facility (RRSAF) or if offloading is enabled.

If you offload column-level processing for a z/OS data source to the Linux, UNIX, or Windows system where the Integration Service runs, PowerExchange uses the **Map Location User** and **Map Location Password** values that are specified on the connection to control access to all resources. This connection is a PWX NRDB CDC application connection or PWX DB2zOS CDC application connection for which offload processing is enabled.

- **Capture registration access.** Set the first parameter in the SECURITY statement to 2 to require a valid z/OS user ID and password that has READ access to the CAPX.REG.* resource profiles to control user access to capture registrations. If you specify another option, your z/OS security product controls access to capture registrations at the data set level only.
- **Extraction map access.** Set the first parameter in the SECURITY statement to 2 to require a valid z/OS user ID and password that has READ access to the data set that is defined in the DTLCAMAP DD statement of the PowerExchange Listener JCL to control user access to extraction maps.
- **Data map access.** Set the first parameter in the SECURITY statement to 2 and enter DM_SUBTASK=Y in the DBMOVER configuration file to have PowerExchange use FACILITY class profiles to control user access to data maps. If you specify another option, your z/OS security product controls access to data maps at the data set level only.
- **PowerExchange Listener commands.** Set the first parameter in the SECURITY statement to 2 to have PowerExchange use FACILITY class profiles to control user access to PowerExchange Listener commands issued from the PowerExchange Navigator or DTLUTSK utility. If you specify another option, PowerExchange does not control access to commands issued from the PowerExchange Navigator or the DTLUTSK utility.
- **Source database access for change capture.** To capture data, the z/OS ECCRs must meet database-specific security requirements and run under a valid z/OS user ID and password that passes PowerExchange Listener security checking.

- **z/OS data access for remote PowerExchange Logger for Linux, UNIX, and Windows logging.** If you log data from z/OS data sources to remote PowerExchange Logger for Linux, UNIX, and Windows log files, set the SECURITY option to 2 in the DBMOVER configuration file on the z/OS system. Ensure that the user ID and password in the PowerExchange Logger for Linux, UNIX, and Windows configuration file, pwxocl, is a valid z/OS user ID and password that can pass z/OS security checking. Also, to access capture registrations, ensure that this user ID and password has READ access to the CAPX.REG.* resource profiles in the FACILITY class.
- **Adabas file write access.** In PowerExchange data maps, you can specify passwords for Adabas files. Set the first parameter in the SECURITY statement to 2. PowerExchange uses FACILITY class profiles to control write access to Adabas files. Otherwise, PowerExchange does not control write access to Adabas files.
- **Datacom table read access.** Set the first parameter in the SECURITY statement to 2 to have PowerExchange use FACILITY class profiles to control read access to Datacom tables. Otherwise, PowerExchange does not control read access to Datacom tables.
- **DB2 for z/OS access.** Set the first parameter in the SECURITY statement to 2 and enter MVSDB2AF=RRSAF in the DBMOVER configuration member to have PowerExchange use the connection user ID to access DB2 resources. Otherwise, PowerExchange uses the user ID under which the PowerExchange Listener runs.
- **IMS database write access.** Set the first parameter in the SECURITY statement to 2 to have PowerExchange use FACILITY class profiles to control write access to IMS databases. Otherwise, PowerExchange does not control write access to IMS databases.
- **Authorization for PowerExchange Agent services and commands.** Set the InitAuthCheck parameter to YES in the AGENTCTL parameter file to have PowerExchange authorize user requests to initialize PowerExchange Agent services or issue PowerExchange Agent commands. For more information, see the *PowerExchange CDC Guide for z/OS*.
- **User authentication for the pwxcmd program.** Set the first parameter in the SECURITY statement to 1 or 2 to have PowerExchange use operating system facilities to authenticate users of the pwxcmd program. If you configure PowerExchange selective sign-on, PowerExchange checks operating system user IDs and passwords after successful selective sign-on checking.
- **Authorization for running pwxcmd commands.** Set the first parameter in the SECURITY statement to 2 on the system that is the target of a command. PowerExchange checks resource profiles to determine whether the user ID supplied on the pwxcmd program is authorized to run commands. Otherwise, authority to run pwxcmd commands is not checked.
- **Selective sign-on.** Set the second parameter in the SECURITY statement to Y to have PowerExchange use the selective sign-on file to limit the users that connect to PowerExchange. Otherwise, any operating system user ID can connect to PowerExchange.

RELATED TOPICS:

- [“User Authentication for the pwxcmd Program on z/OS” on page 277](#)
- [“Authorization to Run pwxcmd Commands on z/OS” on page 278](#)
- [“PowerExchange Selective Sign-on” on page 293](#)

Resource Access Requirements on z/OS

If you set the first parameter in the SECURITY statement to 2 in the DBMOVER configuration member on z/OS, PowerExchange user credentials must pass z/OS security checking.

The user ID must have access to specific PowerExchange resources that are required to perform the PowerExchange function.

For example, users might need the following levels of access:

- To read change data from the source, users need READ access to CAPX.REG.* resources profiles for the capture registrations.
- To extract change data during PowerCenter CDC sessions, users need READ access to the data set defined in the DTLCAMAP DD statement of the PowerExchange Listener JCL to access extraction maps.
- If the PowerExchange Listener uses a data set for the PowerExchange message log file specified in the DTLLOG DD or DTLLOGnn DD statement, users must have UPDATE access to that file.
- To create, update, or delete data maps, users must have UPDATE access to the VSAM data set specified in the DATAMAP DD statement of the PowerExchange Listener JCL.

Data Set Access Requirements for PowerExchange Jobs

The user ID under which PowerExchange jobs and started tasks run must have the proper authority to access the data sets that PowerExchange allocates for bulk data movement and CDC processing.

PowerExchange jobs and started tasks use DD statements in their JCL to allocate data sets. They also dynamically allocate data sets during execution.

The following table lists the access authority that the PowerExchange Listener and netport jobs require for bulk data processing:

DD Name	PowerExchange Listener Authority	Netport Jobs Authority
STEPLIB	READ	READ
DATAMAP	UPDATE	READ
DTLCFG	READ	READ
DTLKEY	READ	READ
DTLLOG	UPDATE	UPDATE
DTLLOGnn	UPDATE	UPDATE
DTLMSG	READ	READ
DTLSGN	READ	READ

The following table lists the access authority that PowerExchange jobs and started tasks require for CDC processing:

DD Name	PowerExchange Listener Authority	PowerExchange Agent Authority	PowerExchange Logger Authority	PowerExchange Condense Authority
STEPLIB	READ	READ	READ	READ
DATAMAP	UPDATE	-	-	READ
DTLAMCPR	UPDATE	-	-	READ
DTLCACDC	READ	-	-	UPDATE

DD Name	PowerExchange Listener Authority	PowerExchange Agent Authority	PowerExchange Logger Authority	PowerExchange Condense Authority
DTLCACDE	UPDATE	-	-	READ
DTLCACFG	N/A	-	-	READ
DTLCAMAP	UPDATE	-	-	READ
DTLCFG	READ	READ	-	READ
DTLKEY	READ	READ	-	READ
DTLLOG	UPDATE	UPDATE	-	UPDATE
DTLLOGnn	UPDATE	UPDATE	-	UPDATE
DTLMSG	READ	READ	-	READ
DTLSGN	READ	-	-	READ
EDMMSG Note: Dynamically allocated SYSOUT data set.	UPDATE	-	UPDATE	UPDATE
EDMPARMS	READ	READ	READ	READ
EDMSCTL	-	READ	-	-
EDMSLOG Note: Dynamically allocated SYSOUT data set.	-	UPDATE	-	-
ERDS01	-	-	UPDATE	-
ERDS02	-	-	UPDATE	-

The following table lists the access authority that PowerExchange CDC jobs and started tasks require for data sets that are dynamically allocated to system-generated DD names:

Job or Started Task	Data Sets	Authority Required
PowerExchange Agent	CCT data set, which contains capture registrations	UPDATE
PowerExchange Logger	PRILOG and SECLOG log data sets	UPDATE
PowerExchange Logger	Archive log data sets	ALTER
PowerExchange Condense	Checkpoint data sets	ALTER
PowerExchange Condense	Condense files	ALTER

Job or Started Task	Data Sets	Authority Required
IMS log-based ECCR	The RECON data sets that the IMS log-based ECCR processes	READ
IMS log-based ECCR	IMS system log data sets (SLDS) and IMS DL/1 batch log data sets (//IEFRDR) contained in the RECON data sets that the IMS log-based ECCR processes	READ

RELATED TOPICS:

- [“Data Set Level Security for Capture Registrations and Extraction Maps” on page 269](#)

Data Set Access Requirements for PowerExchange ECCRs

You must authorize access to the data sets used by PowerExchange ECCRs to the user ID under which the ECCR run.

PowerExchange ECCRs allocate data sets by using DD statements in JCL, and also dynamically allocate data sets during execution.

The following table lists the authority that the Adabas, DB2, and IMS ECCRs require for data sets, by DD name:

DD Name	Adabas ECCR Authority	Adabas User Exit 2 Job Authority	DB2 ECCR Authority	IMS Log-Based ECCR Authority	IMS Synchronous ECCR Authority
STEPLIB	READ	READ	READ	READ	READ
DATAMAP	READ	-	-	READ	-
DDASSOR1	UPDATE	-	-	-	-
DDCARD	READ	-	-	-	-
DDDATAR1	UPDATE	-	-	-	-
DDWORKR1	UPDATE	-	-	-	-
DTLADKSD	READ	-	-	-	-
DTLAMCPR	READ	-	-	READ	-
DTLCACDC	-	-	-	READ	-
DTLCACFG	READ	-	-	READ	-
DTLCACHG	-	-	-	READ	-
DTLCCADA	-	UPDATE	-	-	-
DTLCCPLG	-	READ	-	-	-
DTLCFG	READ	READ	-	READ	-

DD Name	Adabas ECCR Authority	Adabas User Exit 2 Job Authority	DB2 ECCR Authority	IMS Log-Based ECCR Authority	IMS Synchronous ECCR Authority
DTLKEY	READ	READ	-	READ	-
DTLLOG	UPDATE	UPDATE	-	UPDATE	-
DTLMSG	READ	READ	-	READ	-
EDMPARMS	READ	-	READ	READ	READ
REPL2CTL	-	-	READ	-	-
REPL2OPT	-	-	READ	-	-

The following table lists the access authority that the batch VSAM and CICS/VSAM ECCRs require for data sets, by DD name:

DD Name	Batch VSAM ECCR Authority	CICS/VSAM ECCR Authority
STEPLIB	READ	READ
DFHRPL	-	READ
DTLAMCPR	-	-
DTLCACFG	-	-
DTLCFG	-	-
DTLLOG	-	-
DTLLOGnn	-	-
DTLMSG	-	-
EDMPARMS	READ	READ

The following table lists the access authority that the Datacom table-based and IDMS log-based ECCRs require for data sets, by DD name:

DD Name	Datacom Table-Based	IDMS Log-Based
STEPLIB	READ	READ
DTLAMCPR	READ	READ
DTLCACFG	READ	READ
DTLCFG	READ	READ
DTLLOG	UPDATE	UPDATE

DD Name	Datacom Table-Based	IDMS Log-Based
DTLLOGnn	UPDATE	UPDATE
DTLKEY	READ	READ
DTLMSG	READ	READ
EDMPARMS	READ	READ

Security for Capture Registrations and Extraction Maps on z/OS

On a z/OS system, you control access to capture registrations and extraction maps at both the data set and RACF class levels.

For some types of access, such as the ability to delete a capture registration, users require UPDATE access at the data set and RACF class levels. For other types of access, such as the ability to extract data, users require READ access at the data set and RACF class levels.

To control access at the data set level, grant users READ or UPDATE access to the data sets assigned to the DTLCAMAP and DTLAMCPR DD names in the PowerExchange Listener JCL.

To control access at the RACF class level, specify 2 in the first parameter of the SECURITY statement in the DBMOVER configuration file, create CAPX.REG.* and CAPX.CND.* resource profiles, and grant users the appropriate level of access to these profiles.

Data Set Level Security for Capture Registrations and Extraction Maps

You can control access to capture registrations and extraction maps at the data set level.

By granting users certain levels of access to the data sets assigned to the DTLAMCPR and DTLCAMAP DD names in the PowerExchange Listener JCL, you can authorize users to extract data and view, add, edit, and delete capture registrations and extraction maps.

Note: To provide additional security for some of these tasks, specify 2 in the first parameter of the SECURITY statement, create CAPX.REG.* and CAPX.CND.* resource profiles, and grant users the appropriate level of access to these profiles. For more information, see [“RACF Class Security for Capture Registrations and Extractions Maps” on page 270](#).

Control access to capture registrations and extraction maps at the data set level, as follows:

- To authorize users to view capture registrations, grant users UPDATE access to the data set assigned to the DTLAMCPR DD name in the PowerExchange Listener JCL.
- To authorize users to add, edit, or delete capture registrations, grant users UPDATE access to the data set assigned to the DTLAMCPR DD name in the PowerExchange Listener JCL.
- To authorize users to view extraction maps, grant users READ access to the data set assigned to the DTLCAMAP DD name in the PowerExchange Listener JCL.
- To authorize users to extract change data, grant users READ access to the data set assigned to the DTLCAMAP DD name in the PowerExchange Listener JCL.
- To authorize users to add, edit, and delete extraction maps, grant users UPDATE access to the data sets assigned to the DTLCAMAP and DTLAMCPR DD names in the PowerExchange Listener JCL.

RELATED TOPICS:

- [“Data Set Access Requirements for PowerExchange Jobs” on page 265](#)
- [“RACF Class Security for Capture Registrations and Extractions Maps” on page 270](#)

RACF Class Security for Capture Registrations and Extractions Maps

PowerExchange checks PowerExchange-specific resource profiles to determine access to capture registrations and extraction maps.

To enable RACF class security checking, specify 2 in the first parameter of SECURITY statement in the DBMOVER configuration file and create and grant users the appropriate level of access to resource profiles.

PowerExchange uses the MVS System Authorization Facility (SAF) interface to check the following resource profiles:

- CAPX.REG.* resource profiles control access to capture registrations, including the ability to add and edit capture registrations.
- CAPX.CND.* resource profiles control the extraction of the data, but not the ability to add, edit, and delete extraction maps.

By default, PowerExchange checks CAPX.REG.* and CAPX.CND.* resource profiles in the FACILITY class. Use the RACF_CLASS parameter in the DBMOVER configuration file to specify a different class for checking security profiles. If you specify a different class, you must define it in RACF or another security product by using resource name with maximum length of 8 bytes.

Control access to capture registrations and extraction maps at the RACF class level, as follows:

- To authorize users to view capture registrations, specify 2 in the first parameter of the SECURITY statement and create and grant users READ access to CAPX.REG.* resource profiles.
- To authorize users to add, edit, or delete capture registrations, specify 2 in the first parameter of the SECURITY statement and create and grant users UPDATE access to CAPX.REG.* resource profiles.
- To authorize users to extract change data, specify 2 in the first parameter of the SECURITY statement and create and grant users READ access to CAPX.CND.* resource profiles.
- To authorize users to add, edit, or delete extraction maps, specify 2 in the first parameter of the SECURITY statement and create and grant users READ access to CAPX.REG.* resource profiles.

Restriction: If you specify 2 in the first parameter of SECURITY statement, you must create resource profiles for capture registrations and extraction maps. Otherwise, PowerExchange denies access to these resources.

Resources Profiles for Capture Registrations and Extraction Maps

To authorize users to extract data, view capture registrations, and add, edit, and delete capture registrations and extraction maps, create and grant users the appropriate level of access to resource profiles.

For capture registrations, a resource profile has the following form:

```
CAPX.REG.dbtype.dbid.registration_name
```

The following table describes the variables in a capture registration resource profile:

Variable	Description
<i>dbtype</i>	The database type, which is one of the following: <ul style="list-style-type: none">- ADA. Adabas.- DB2. DB2.- DCM. Datacom.- IDM. IDMS.- IMS. IMS.- VSM. VSAM.
<i>dbid</i>	The instance, database ID, or collection ID specified in the registration group that contains the capture registration. Valid value is a character string.
<i>registration_name</i>	The name of the capture registration. Valid value is a character string.

For extraction maps, a resource profile has the following form:

```
CAPX.CND.dbid.extraction_map_name
```

The following table describes the variables in an extraction map resource profile:

Variable	Description
<i>dbid</i>	The instance, database ID, or collection ID that you specified in the extraction group that contains the extraction map. Valid value is a character string.
<i>extraction_map_name</i>	The name of the extraction map. Valid value is a character string.

You can define generic resource profiles to minimize the number of definitions. For example, the following generic profile in RACF covers all capture registration and extraction map profiles:

```
CAPX.**
```

Granting Access to Capture Registrations and Extraction Maps

To authorize users to extract change data and view, add, edit, and delete capture registrations and extraction maps, grant users the appropriate access at the data set level and RACF class level.

Prerequisite: Before granting access, verify that the data set and resource profiles are closed. To do this, create a rule for the data set and the CAPX.* resource profile with a UACC of (NONE).

The following table lists the required data set access authority and resource profile access authority for capture registrations:

Task	Required Data Set Access Authority	Required Resource Profile Access Authority
View capture registrations.	UPDATE access to the data set assigned to the DTLAMCPR DD name in the PowerExchange Listener JCL	READ access to the CAPX.REG.* resource profile
Add capture registrations. Edit capture registrations to change selected columns, registration status, or condense type. Delete capture registrations.	UPDATE access to the data set assigned to the DTLAMCPR DD name in the PowerExchange Listener JCL	UPDATE access to the CAPX.REG.* resource profile

The following table lists the required data set access authority and resource profile access authority for extraction maps:

Task	Required Data Set Access Authority	Required Resource Profile Access Authority
View extraction maps.	READ access to the data set assigned to the DTLCAMAP DD name in the PowerExchange Listener JCL	None
Extract change data.	READ access to the data set assigned to the DTLCAMAP DD name in the PowerExchange Listener JCL	READ access to the CAPX.CND.* resource profile
Add extraction maps. Edit extraction maps. Delete extraction maps.	<ul style="list-style-type: none"> UPDATE access to the data set assigned to the DTLAMCPR DD name in the PowerExchange Listener JCL UPDATE access to the data set assigned to the DTLCAMAP DD name in the PowerExchange Listener JCL 	READ access to the CAPX.REG.* resource profile

Security for Capture Registrations and Extraction Maps - Examples

You can grant data set and RACF class level authority for different types of users.

A *super user* can add and edit capture registrations and extraction maps, run database row tests against extraction maps to preview data, and extract change data. To permit a user to complete these tasks, grant the user all the following levels of access:

- UPDATE access to the data set assigned to the DTLAMCPR DD name in the PowerExchange Listener JCL
- UPDATE access to the data set assigned to the DTLCAMAP DD name in the PowerExchange Listener JCL
- UPDATE access to the CAPX.REG.* resource profile
- READ access to the CAPX.CND.* resource profile

To permit a user to view capture registrations and extract change data, including running a database row test against an extraction map and running workflow extractions, grant the user all the following levels of access:

- UPDATE access to the data set assigned to the DTLAMCPR DD name in the PowerExchange Listener JCL

Note: This level of access does not grant the user the ability to create or amend capture registrations.

- READ access to the data set assigned to the DTLCAMAP DD name in the PowerExchange Listener JCL
- READ access to the CAPX.REG.* resource profile
- READ access to the CAPX.CND.* resource profile

For example, to permit a user to run a database row test against an extraction map name with a prefix of CIW* for data captured from DB2 subsystem DB2P, grant the user READ access to the following resource profile:

```
CAPX.CND.DB2P.CIW*
```

To permit a user to run a database row test against an extraction map and run workflow extractions, but not view capture registrations, grant the user all the following levels of access:

- READ access to the data set assigned to the DTLCAMAP DD name in the PowerExchange Listener JCL
- UPDATE access to the data set assigned to the DTLAMCPR DD name in the PowerExchange Listener JCL
- READ access to the CAPX.CND.* resource profile

Security for Data Maps

PowerExchange checks PowerExchange-specific resource profiles to determine whether to permit access to data maps. For this security checking to occur, you must specify the following parameters in the DBMOVER configuration file:

- 2 in the first parameter of SECURITY statement
- Y for the DM_SUBTASK parameter

PowerExchange then uses the MVS System Authorization Facility (SAF) interface to check access to the resource profiles.

By default, PowerExchange checks resource profiles in the FACILITY class. Use the RACF_CLASS parameter in the DBMOVER configuration file to specify a different class for checking security profiles. If you specify a different class, it must be defined in RACF or the security product you use, with a resource name that has a maximum length of 8 bytes.

Restriction: If you specify 2 in the first parameter of SECURITY statement and Y for the DM_SUBTASK parameter, you must specify a resource profile for data maps. Otherwise, PowerExchange fails access.

Resource Profiles for Data Maps

Configure resource profiles for data maps and set the access authorities.

The resource profile for access to data maps has the following form:

```
DTL.DATAMAP.resource_name
```

The *resource_name* variable specifies the suffix for the resource profile. This suffix is the value specified for the DM_RESOURCE parameter in the DBMOVER configuration file, or if DM_RESOURCE is not specified, the default of DATASET.

The following table describes the types of access authorities that you can include:

Access Type	Description
READ	Allows users to read data maps
UPDATE	Allows users to define, delete, and modify data maps

If you do not define any resource profiles for data maps, PowerExchange denies access.

You can define generic resource profiles to minimize the number of definitions. For example, the following generic profile in RACF covers all DM_RESOURCE values:

```
DTL.DATAMAP.**
```

Security for PowerExchange Listener Commands

PowerExchange checks PowerExchange-specific resource profiles to determine whether to permit access to PowerExchange Listener commands issued from the PowerExchange Navigator or the DTLUTSK utility. For this security checking to occur, you must specify 2 in the first parameter of SECURITY statement in the DBMOVER configuration file. PowerExchange then uses the MVS System Authorization Facility (SAF) interface to check access to the resource profiles.

By default, PowerExchange checks resource profiles in the FACILITY class. Use the RACF_CLASS parameter in the DBMOVER configuration file to specify a different class for checking security profiles. If you specify a different class, it must be defined in RACF or the security product you use, with a resource name that has a maximum length of 8 bytes.

Restriction: If you specify 2 in the first parameter of SECURITY statement, you must specify a resource profiles for PowerExchange Listener commands. Otherwise, PowerExchange fails PowerExchange Listener commands issued from the PowerExchange Navigator or the DTLUTSK utility.

Resources Profiles for PowerExchange Listener Commands

Configure resource profiles for PowerExchange Listener commands and set the access authorities.

The resource profile for the LISTTASK command has the following form:

```
DTL.TASKCTRL.DISPLAY
```

The resource profile for the STOPTASK command has the following form:

```
DTL.TASKCTRL.STOPTASK
```

Users that have READ access to the resource profiles can use the PowerExchange Navigator or the DTLUTSK utility to issue LISTTASK or STOPTASK commands. Otherwise, access to these commands is denied. If you do not define any resource profiles, PowerExchange denies access to these commands.

You can define generic resource profiles to minimize the number of definitions. For example, the following generic profile in RACF covers all PowerExchange Listener commands that you can issue from the PowerExchange Navigator or the DTLUTSK utility:

```
DTL.TASKCTRL.**
```

Adabas Security

You can use the following Adabas security options with PowerExchange:

- Password security

- SAF security
- Security by Value

To use Adabas password security, specify file passwords in PowerExchange data maps. To use Adabas SAF security or security by value, no action is required in PowerExchange.

Restriction: PowerExchange does not support Natural Security, which is a Natural application. PowerExchange supports Adabas data access directly, not through an application layer.

PowerExchange checks PowerExchange-specific resource profiles to determine whether to permit write access to Adabas files. For this security checking to occur, specify 2 in the first parameter of SECURITY statement in the DBMOVER configuration file. PowerExchange then uses the MVS System Authorization Facility (SAF) interface to check access to the resource profiles.

By default, PowerExchange checks resource profiles in the FACILITY class. Use the RACF_CLASS parameter in the DBMOVER configuration file to specify a different class for checking security profiles. If you specify a different class, it must be defined in RACF or the security product you use, with a resource name that has a maximum length of 8 bytes.

Resource Profiles for Controlling Write Access to Adabas Files

Configure resource profiles for write access to Adabas files and set the access authorities.

The resource profile for controlling write access to Adabas files has the following form:

```
DTL.DBWRITE.ADABAS.DBdbid.FNfile_num
```

Where:

- *dbid* is a physical database ID for the database that contains the file. This value begins with "DB" and is followed by the database ID, which must be left-padded with zeroes so that the ID has five digits. Valid values are 00001 through 99999.
- *file_num* is a file number. This value begins with FN and is followed by the file number, which must be left-padded with zeros so that the file number has five digits. Valid values are from 00001 through 99999.

Users that have UPDATE access to the resource profile can use PowerExchange to write data to the Adabas file. Otherwise, access is denied. If you do not define any resource profiles, PowerExchange does not control write access.

You can define generic resource profiles to minimize the number of definitions. For example, the following generic profile in RACF covers all Adabas files in all databases:

```
DTL.DBWRITE.ADABAS.**
```

The following generic profile in RACF covers all Adabas files in databases 1,000 through 1,999:

```
DTL.DBWRITE.ADABAS.DB01*.**
```

Datacom Security

PowerExchange checks PowerExchange-specific resource profiles to determine whether to permit Datacom data map creation and subsequent read access to Datacom tables. For this security checking to occur, you must specify 2 as the value of the first parameter of the SECURITY statement in the DBMOVER configuration file. PowerExchange then uses the MVS System Authorization Facility (SAF) interface to check access to the resource profiles.

By default, PowerExchange checks resource profiles in the FACILITY class. Use the RACF_CLASS parameter in the DBMOVER configuration file to specify a different class for checking security profiles. If you specify a different class, it must be defined in RACF or the security product you use, with a resource name that has a maximum length of 8 bytes.

RELATED TOPICS:

- [“Security for Capture Registrations and Extraction Maps on z/OS” on page 269](#)

Resource Profiles for Controlling Read Access to Datacom Tables

Configure resource profiles for read access to Datacom table and set the access authorities.

The resource profile for controlling read access to Datacom tables has the following form:

```
DTL.DBREAD.DATACOM.Ddatabase_id.short_table_name
```

Where:

- *database_id* is a database identifier (ID) that begins with “D” and is followed by five digits. Database IDs must be left-padded with zeroes so that the ID always has five digits. PowerExchange matches the database ID in the resource profile against the database IDs of the actual Datacom databases. Valid values are 00001 through 99999.
- *short_table_name* is the short table name that is defined in Datacom for a table. PowerExchange matches the specified short table name against Datacom short table names and indirectly against Datacom long table names to determine the tables for which RACF is to restrict Datacom data map creation or READ access to the tables. You do not need to define separate RACF resource profiles for long table names. If you create resource profiles for long table names only, PowerExchange cannot also match on short table names to identify the tables that RACF is to protect.

Users that have READ access to the resource profile can use PowerExchange to read data from the Datacom table or obtain metadata for the table. Otherwise, access is denied. If you do not define a resource profile, PowerExchange does not control read access. For example, if you have a database ID of 23 and short table name of STL, create the following resource profile:

```
DTL.DBREAD.DATACOM.D00023.STL
```

You can define generic resource profiles to minimize the number of definitions. For example, the following generic profile in RACF covers all Datacom tables in all databases:

```
DTL.DBREAD.DATACOM.**
```

If an unauthorized user attempts a task such as a database row test of a data map in PowerExchange Navigator, PowerExchange issues the following message:

```
PWX-00243 Userid user_id not authorized to access resource
DTL.DBREAD.DATACOM.Ddatabase_id.short_table_name req=R by RACF.
```

In this message, the req value of “R” represents READ access.

If an unauthorized user attempts a task such as creating a data map, PowerExchange issues the following message:

```
PWX-02530 The metadata requested was not available, please check your parameters.
```

DB2 for z/OS Security

To use connection user IDs control access to DB2 tables and resources, you must specify the following statement and parameter in the DBMOVER configuration file on the z/OS system:

- 2 in first parameter of the SECURITY statement
- RRSF for the MVSDB2AF parameter

Otherwise, PowerExchange uses the user ID under which the PowerExchange Listener or netport job runs to control access to DB2 tables and resources.

Note: To specify RRSF for the MVSDB2AF parameter, the Resource Recovery Services (RRS) component of z/OS must be active.

IMS Security

PowerExchange checks PowerExchange-specific resource profiles to determine whether to permit write access to IMS databases. For this security checking to occur, you must specify 2 in the first parameter of SECURITY statement in the DBMOVER configuration file. PowerExchange then uses the MVS System Authorization Facility (SAF) interface to check access to the resource profiles.

By default, PowerExchange checks resource profiles in the FACILITY class. Use the RACF_CLASS parameter in the DBMOVER configuration file to specify a different class for checking security profiles. If you specify a different class, it must be defined in RACF or the security product you use, with a resource name that has a maximum length of 8 bytes.

Resource Profiles for Controlling Write Access to IMS Databases

Configure resource profiles for write access to IMS databases and set the access authorities.

The resource profile for controlling write access to IMS databases access through DL/I batch or BMP jobs has the following form:

```
DTL.DBWRITE.IMS
```

The resource profile for controlling write access to IMS databases access through IMS ODBA has the following form:

```
DTL.DBWRITE.IMS.ims_id
```

The *ims_id* variable specifies the IMS SSID value that you specify on the **Data Maps Properties** dialog box in the PowerExchange Navigator. Although you can also specify an IMS SSID value for DL/1 batch through the PowerExchange Navigator, this value is not used in the resource profile specification for DL/1 batch.

Users that have UPDATE access to the resource profile can use PowerExchange to write data to IMS databases. Otherwise, access is denied. If you do not define a resource profile, PowerExchange does not control write access.

You can define generic resource profiles to minimize the number of definitions. For example, the following generic profile in RACF covers writing to IMS databases with DL/I batch or ODBA:

```
DTL.DBWRITE.IMS.**
```

You can use this type of generic profile to prevent general write access.

User Authentication for the pwxcmd Program on z/OS

You can configure PowerExchange to authenticate operating system user IDs and passwords to control the use of the pwxcmd program.

To verify user credentials for pwxcmd program use, complete the following configuration tasks for each PowerExchange installation that is the target of a command from the pwxcmd program:

- In the DBMOVER configuration member on the system that is the target of the command, set the first parameter of the SECURITY statement to 1 or 2.
- Configure the necessary permissions to authenticate operating system user credentials.

Authorization to Run pwxcmd Commands on z/OS

In addition to authenticating user credentials for pwxcmd program use, you can configure PowerExchange to authorize users to run specific pwxcmd commands.

On the system that is the target of pwxcmd commands, set the first parameter in the SECURITY statement to 2. Then, provide resource profiles on the target system to authorize users to run specific pwxcmd commands.

PowerExchange checks the resource profiles to determine whether the user ID supplied on the pwxcmd program is authorized to run commands.

In the resource profiles, the RACF class is either the default FACILITY class or the value specified in the RACF_CLASS parameter in the DBMOVER configuration file.

To authorize a user to issue pwxcmd commands, grant the user READ access to the appropriate resource profiles. Otherwise, access is denied.

The resource profile that controls access to pwxcmd commands has the following form:

```
DTL.CMD.service_type.service_name.command_name
```

The following table describes the variables in a resource profile for pwxcmd commands:

Variable	Description
service_type	PowerExchange service type. Options are: <ul style="list-style-type: none">- LISTENER. PowerExchange Listener.- CONDENSE. PowerExchange Condense.
service_name	The service name specified in the CMDNODE statement in the DBMOVER configuration file.
command_name	The pwxcmd command. You can issue the following PowerExchange Condense commands through the pwxcmd program: <ul style="list-style-type: none">- condense- displaystatus- fileswitch- shutcond- shutdown You can issue the following PowerExchange Listener commands through the pwxcmd program: <ul style="list-style-type: none">- close- closeforce- listtask- stoptask

Define generic resource profiles to minimize the number of definitions.

For example, the following profile grants access to all PowerExchange Condense commands issued through the pwxcmd program:

```
DTL.CMD.CONDENSE.**
```

The following profile grants access to all PowerExchange Condense commands issued through the pwxcmd program that specify the MY_COND service name:

```
DTL.CMD.CONDENSE.MY_COND.**
```

Note: To authenticate users to run pwxcmd commands, configure PowerExchange to check user credentials for the pwxcmd program.

PowerExchange Passphrases

You can enter a valid PowerExchange passphrase instead of password to access sources and targets on z/OS and i5/OS. Passphrases provide enhanced security because they are longer and can contain a wide range of allowable character types. You can also use encrypted passphrases.

You can enter a passphrase or encrypted passphrase in PowerCenter and the PowerExchange Navigator.

In PowerCenter, enter a passphrase when performing any of the following tasks:

- Defining PWXPC DB2zOS, DB2iOS, and NRDB connections for bulk data movement and CDC
- Defining PowerExchange ODBC connections
- Importing source and target definitions, importing extraction maps, and previewing data with PWXPC
- Importing source and target definitions with PowerExchange ODBC

For more information, see *PowerExchange Interfaces for PowerCenter*.

In the PowerExchange Navigator, enter a passphrase when performing any of the following tasks:

- Adding or viewing registration groups, extraction groups, and application groups
- Adding personal metadata profiles
- Sending data maps to a remote node, or importing remote data maps
- Defining logons for accessing remote data maps and data
- Importing copybooks or i5/OS DDSs from remote locations
- Viewing a remote data file
- Performing a database row test
- Generating an encrypted passphrase

For more information, see the *PowerExchange Navigator User Guide*.

Note: You can also enter PowerExchange passphrases when defining connections to data sources on z/OS or i5/OS in the Informatica Developer tool. For more information, see the *Informatica Developer Tool Guide*.

Passphrase Definition

Passphrases have the following length limits:

- An i5/OS passphrase can be from 9 to 31 characters in length.
- A z/OS passphrase can be from 9 to 128 characters in length (PWXPC connections) or from 9 to 79 characters in length (ODBC connections).

Note: On z/OS, a valid RACF passphrase can be up to 100 characters in length. PowerExchange truncates passphrases longer than 100 characters when passing them to RACF for validation.

Passphrases can contain the following characters:

- Uppercase and lowercase letters
- The numbers 0 to 9
- Spaces
- The following special characters:
- ' - ; # \ , . / ! % & * () _ + { } : @ | < > ?

Note: The first character is an apostrophe.

Passphrases cannot contain single quotation marks ('), double quotation marks ("), or currency symbols.

Do not enclose passphrases in quotation marks.

Passphrase Usage Considerations

The following usage considerations apply to passphrases:

- All PowerExchange instances in your environment must use PowerExchange version 9.6.0 or later.
- On z/OS, passphrases are not supported for netport jobs. Passphrases also are not supported for JCL template jobs such as DB2LDJCL and IDMSMJCL in the RUNLIB library that are submitted by the Listener in response to a PowerCenter or PowerExchange Navigator client request for connection to a source or target.
- The PowerExchange Listener must run with a security setting of SECURITY=(1,N) or higher in the DBMOVER member.
- To use passphrases for IMS connections, ensure that the following requirements are met:
 - You must configure ODBA access to IMS as described in the *PowerExchange Navigator User Guide*.
 - You must use IMS data maps that specify IMS ODBA as the access method. Do not use data maps that specify the DL/1 BATCH access method because this access method requires the use of netport jobs, which do not support passphrases.
 - The IMS database must be online in the IMS control region to use ODBA access to IMS.

PowerExchange LDAP User Authentication

You can use LDAP user authentication to authenticate client requests to a PowerExchange Listener or PowerExchange Logger for Linux, UNIX, and Windows instance that runs on a Linux, UNIX, or Windows system.

If you enable LDAP authentication, the PowerExchange Listener or PowerExchange Logger connects to an LDAP server to authenticate the LDAP enterprise user ID and password of the client that requests a connection to the PowerExchange Listener or PowerExchange Logger.

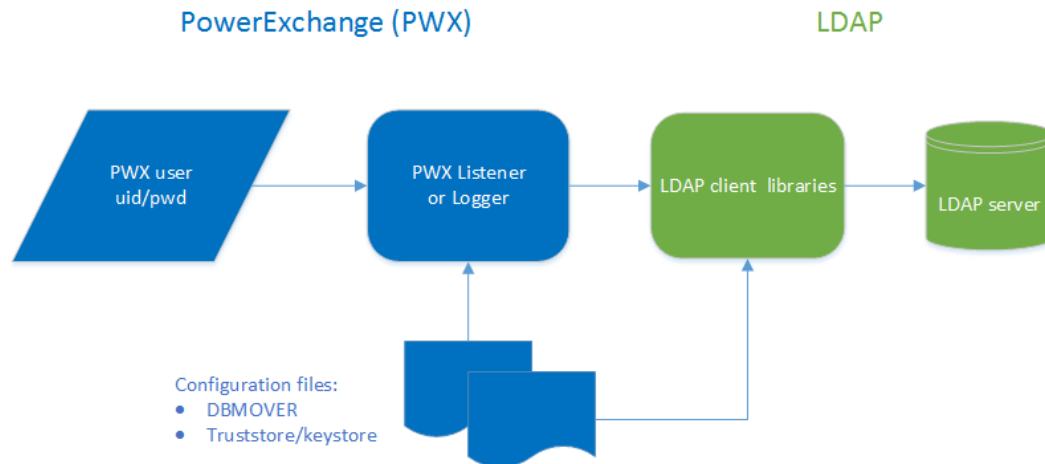
LDAP authentication in PowerExchange provides the following features:

- LDAP user validation. PowerExchange can validate PowerExchange user credentials against an entry in LDAP.
- Flexible search. DBMOVER configuration statements provide the ability to specify multiple LDAP search locations, search filters, search tree depth, and a login attribute to key against the PowerExchange user ID.
- Failover. You can configure a list of LDAP servers in order of priority for failover, so that if a higher priority server is down, subsequent calls fail over to another server in the list.
- Relational pass-through authentication. You can configure pass-through authentication for relational connections. Pass-through authentication delegates authentication to the underlying relational database.
- LDAP with TLS. LDAP user authentication in PowerExchange supports the Transport Layer Security (TLS) and Secure Sockets Layer (SSL) protocols to protect against snooping, tampering, and man-in-the-middle security threats.

PowerExchange LDAP Authentication Components

LDAP authentication in PowerExchange involves PowerExchange and LDAP components.

The following image shows the main components that are involved in LDAP user authentication in PowerExchange:



PowerExchange Listener or Logger

The PowerExchange Listener or the PowerExchange Logger for Linux, UNIX, and Windows receives the credentials of the PowerExchange user from the client program and issues the following calls to the LDAP server:

1. Bind the search user. The PowerExchange Listener or PowerExchange Logger uses the search user to connect to the LDAP server.
2. Search for the PowerExchange user.
3. Bind the PowerExchange user.

The following PowerExchange Listener clients can use LDAP authentication:

- PowerCenter workflow or mapping with a PowerExchange source or target
- PowerExchange Navigator
- pwxcmd commands that connect to the PowerExchange Listener through the SVCNODE port

The following PowerExchange Logger clients can use LDAP authentication:

- pwxcmd commands that connect to the PowerExchange Logger through the SVCNODE port

PowerExchange user credentials

A client program sends the user ID and password of the PowerExchange user to the PowerExchange Listener or PowerExchange Logger.

Configuration files

The PowerExchange Listener or PowerExchange Logger reads LDAP configuration information from the DBMOVER configuration file.

If you configure PowerExchange to use the Transport Layer Security (TLS) protocol to encrypt communications between PowerExchange and the LDAP server, PowerExchange uses keystore and truststore files to authenticate the LDAP client or server.

LDAP client libraries

The LDAP client libraries provide an interface between the PowerExchange Listener or Logger and the LDAP server.

LDAP server

The LDAP server receives and executes the calls that the PowerExchange Listener or PowerExchange Logger issues to bind the search user, search for the PowerExchange user, and bind the PowerExchange user.

Supported Environments for LDAP Authentication

PowerExchange supports the following LDAP implementations:

- OpenLDAP
- Oracle LDAP

For each implementation, PowerExchange supports a different set of platforms, LDAP client libraries, and LDAP servers.

Supported Platforms For LDAP Authentication

To perform LDAP authentication, the PowerExchange Listener or PowerExchange Logger for Linux, UNIX, or Windows must run on a supported platform.

PowerExchange supports the OpenLDAP implementation on the following platforms:

- AIX
- Linux x64
- Solaris SPARC
- Windows x64

PowerExchange supports the Oracle LDAP implementation on the following platforms:

- Linux x64
- Solaris SPARC

Supported LDAP Client Libraries

PowerExchange supports the client libraries for the OpenLDAP and Oracle LDAP implementations. The client libraries must be installed on the PowerExchange Listener or PowerExchange Logger machine.

The PowerExchange installation program installs the OpenLDAP client libraries.

If you are using the Oracle LDAP implementation, you must install the Oracle LDAP client libraries.

Supported LDAP Servers

PowerExchange supports LDAP servers for the OpenLDAP and Oracle LDAP implementations.

For the OpenLDAP implementation, PowerExchange supports the following LDAP servers:

- Microsoft Active Directory
- OpenLDAP
- Apache Directory Server

For the Oracle LDAP implementation, PowerExchange supports the following LDAP server:

- Oracle Directory Server Enterprise Edition (ODSEE) 11.1.1.7.0

LDAP Search User

LDAP servers can be configured with a level of access control that prevents unauthorized access to directory data. As a result, to connect to an LDAP server and search for a PowerExchange user, the PowerExchange Listener or PowerExchange Logger for Linux, UNIX, and Windows uses a search user. A search user is an LDAP user that has access permissions that allow connection to the LDAP server and searching of the relevant LDAP entries. This search user requires credentials so that the LDAP server can authenticate and authorize the connection and search.

To configure the LDAP credentials of the search user in PowerExchange, define the LDAP_BIND_DN statement and the LDAP_BIND_EPWD or LDAP_BIND_PWD statement in the DBMOVER configuration file.

LDAP servers can also be configured for anonymous access. To support this case, define the LDAP_BIND_DN and LDAP_BIND_PWD values to be the empty string.

If you configure LDAP TLS to require client certification and specify a SASL mechanism of EXTERNAL, you do not need to define the LDAP_BIND_DN, LDAP_BIND_EPWD, or LDAP_BIND_PWD statements. In this case, the identity and credentials of the search user are obtained from a client certificate and the LDAP server certificate mapping configuration.

RELATED TOPICS:

- [“LDAP_BIND_DN Statement” on page 134](#)
- [“LDAP_BIND_EPWD Statement” on page 135](#)
- [“LDAP_BIND_PWD Statement” on page 136](#)

Controlling the LDAP Search

The entries within an LDAP directory can be structured in different ways, depending on the needs of the enterprise.

To optimize search performance and to control which areas of the LDAP tree are searched and which LDAP attribute is used as a search key, define the appropriate statements in the DBMOVER configuration file.

RELATED TOPICS:

- [“LDAP_BASE Statement” on page 134](#)
- [“LDAP_SCOPE Statement” on page 140](#)
- [“LDAP_FILTER Statement” on page 137](#)
- [“LDAP_LOGIN_ATTRIBUTE Statement” on page 138](#)
- [“LDAP_SEARCH_TIMEOUT Statement” on page 141](#)
- [“LDAP_BIND_TIMEOUT Statement” on page 136](#)

LDAP Transport Layer Security

You can configure PowerExchange to use the Transport Layer Security (TLS) protocol to encrypt communications between PowerExchange and the LDAP server. You can supplement TLS security with Simple Authentication and Security Layer (SASL) security.

TLS uses encryption to protect against snooping and tampering with network traffic. You can configure TLS to require server authentication only or both client and server authentication. The client in these transactions is the PowerExchange Listener or PowerExchange Logger machine, and the server is the LDAP server.

You can also configure PowerExchange to use the StartTLS extension for secured communications.

The following statements in the DBMOVER configuration file control certificate-based LDAP security:

- The LDAP_OPENSSL statement controls certificate-based LDAP security for an OpenLDAP implementation.
- The LDAP_TLS statement controls certificate-based LDAP security for an Oracle LDAP implementation.
- The LDAP_SASL_MECH statement specifies the authentication mechanism that the Simple Authentication and Security Layer (SASL) uses for either the OpenLDAP or Oracle LDAP implementation.

The OpenLDAP and Oracle LDAP implementations use different keystore formats. OpenLDAP uses OpenSSL certificate and key files, which must be in PEM format. Oracle LDAP uses NSS certificate and key files.

Note: Unless otherwise noted, the term *TLS* is used to denote both the Secure Sockets Layer (SSL) and TLS protocols.

Levels of LDAP Security

You can configure the following levels of LDAP security:

Level 1 LDAP security: Secure LDAP with encryption and server authentication

Level 1 LDAP security uses TLS to encrypt network traffic. Using TLS mitigates snooping and tampering threats.

During a call to the LDAP server, the LDAP server provides a server certificate that proves its identity to PowerExchange. Providing a server certificate mitigates the threat of server spoofing.

For the client to be able to verify the server certificate, a suitable entry must be configured in the truststore on the PowerExchange Listener machine. This entry must provide a chain of trust that enables PowerExchange to verify the trustworthiness of the LDAP server.

Level 2 LDAP security: Secure LDAP with encryption and server and client authentication

Level 2 LDAP security provides the same protections as Level 1 and also provides client authentication.

During a call to the LDAP server, PowerExchange provides a client certificate that proves its identity to the LDAP server. Providing a client certificate mitigates the threat of client spoofing.

For the LDAP server to be able to verify the client certificate, a suitable entry must be configured in the LDAP server truststore. This entry must provide a chain of trust that enables the LDAP server to verify the trustworthiness of the client.

Level 3 LDAP security: Secure LDAP with SASL EXTERNAL authentication

Level 3 LDAP security provides the same protections as Level 2 and also uses the SASL EXTERNAL mechanism.

If you do not use the SASL EXTERNAL mechanism, you must provide search user credentials by defining the LDAP_BIND_DN statement and either the LDAP_BIND_PWD or LDAP_BIND_EPWD statement in the DBMOVER configuration file. Level 3 LDAP security eliminates the need to define these statements.

This configuration requires a special certificate-mapping configuration on the LDAP server so that information from the PowerExchange client certificate can be used to identify the search user in LDAP.

For examples of how to configure the three security levels, see [“Example DBMOVER LDAP Configurations” on page 288](#).

RELATED TOPICS:

- [“LDAP_SASL_MECH Statement” on page 140](#)
- [“LDAP_TLS Statement” on page 141](#)
- [“LDAP_OPENSSL Statement” on page 138](#)

LDAP StartTLS Extended Operation

LDAP servers are usually configured for both plain text (unencrypted) and secured (encrypted) communication. Conventionally, port 389 is dedicated to plain requests and port 636 is dedicated to secured requests. The StartTLS extension allows an application to make both plain text and secured requests on a single port that is normally used for plain text communications. Using StartTLS avoids the need for a dedicated secured port.

You can configure PowerExchange to use the StartTLS extension with any of the three levels of LDAP security.

To enable use of the StartTLS extension, include the `START_TLS=Y` parameter in the `LDAP_TLS` or `LDAP_OPENSSL` statement in the DBMOVER configuration file. Also, change any port settings to use the plain text port.

For examples of how to configure StartTLS for the three security levels, see [“Example DBMOVER LDAP Configurations” on page 288](#).

RELATED TOPICS:

- [“LDAP_OPENSSL Statement” on page 138](#)
- [“LDAP_TLS Statement” on page 141](#)

LDAP Server Failover

You can define multiple LDAP servers. When PowerExchange issues an LDAP authentication request, it connects to the first available LDAP server, based on the order in which the `LDAP_HOST` statements occur in the DBMOVER configuration file.

If a failed server returns to operation, it is used for subsequent LDAP user validation calls.

To define multiple LDAP servers, in the DBMOVER configuration file on the PowerExchange Listener or PowerExchange Logger for Linux, UNIX, and Windows machine, define an `LDAP_HOST` statement for each LDAP server.

Note: PowerExchange supports a basic level of resilience to LDAP server failure by connecting to the first LDAP server for which a connection is available. However, this implementation does not provide resilience to LDAP server failures that do not manifest themselves as connection failures.

RELATED TOPICS:

- [“LDAP_HOST Statement” on page 137](#)

Configuration Tasks

To set up LDAP user authentication in PowerExchange, perform the following LDAP and PowerExchange configuration tasks:

1. If you are using the Oracle LDAP implementation, Install the Oracle LDAP client libraries on each machine where the PowerExchange Listener or PowerExchange Logger for Linux, UNIX, and Windows runs.

For more information, see the documentation for your LDAP software.

Note: If you are using the OpenLDAP implementation, the PowerExchange installation program installs the OpenLDAP client libraries.

2. Set environment variables to enable PowerExchange to locate the LDAP client libraries.

For more information, see [“Setting Environment Variables” on page 286](#).

3. To use TLS with LDAP, configure the required truststore and keystore files.

For more information, see the TLS documentation for your system.

4. Add the appropriate LDAP statements to the PowerExchange DBMOVER configuration file on each PowerExchange Listener or PowerExchange Logger machine that supports LDAP authentication.

For more information, see [“Summary of DBMOVER Statements for LDAP Configuration” on page 286](#).

Setting Environment Variables

To ensure that the PowerExchange Listener or PowerExchange Logger can locate the required Oracle LDAP client libraries, set environment variables.

Note: If you use the OpenLDAP implementation, you do not need to set environment variables.

On a Linux machine, you might enter the following command:

```
export LD_LIBRARY_PATH=/mydir/dsee7/lib64/private:$LD_LIBRARY_PATH
```

On a Solaris SPARC machine, you might enter the following commands:

```
export LD_LIBRARY_PATH=/mydir/dsee7/lib/sparcv9/private:$LD_LIBRARY_PATH
unset LD_LIBRARY_PATH_64
```

Also, to use LDAP commands such as `ldapsearch` or `certutil`, set the `PATH` environment variable.

For example, on a Linux machine, you might enter the following command:

```
export PATH=/mydir/dsee7/dsrk/bin:/mydir/dsee7/bin:$PATH
```

Summary of DBMOVER Statements for LDAP Configuration

To configure PowerExchange for LDAP authentication, define the appropriate statements in the DBMOVER configuration file on a PowerExchange Listener or PowerExchange for Linux, UNIX, and Windows machine that supports LDAP authentication.

The following DBMOVER statements apply to LDAP authentication:

LDAP_BASE

Specifies the location within an LDAP directory from which to conduct a PowerExchange user search.

For more information, see [“LDAP_BASE Statement” on page 134](#).

LDAP_BIND_DN

Specifies the distinguished name (DN) of an LDAP user with sufficient access rights to connect to LDAP and conduct a search for the PowerExchange user.

For more information, see ["LDAP_BIND_DN Statement" on page 134](#).

LDAP_BIND_EPWD

Specifies the encrypted password for the LDAP search user.

For more information, see ["LDAP_BIND_EPWD Statement" on page 135](#).

LDAP_BIND_PWD

Specifies the password for the LDAP search user.

For more information, see ["LDAP_BIND_PWD Statement" on page 136](#).

LDAP_BIND_TIMEOUT

Specifies the number of seconds that a PowerExchange Listener or PowerExchange Logger waits for an LDAP bind request.

For more information, see ["LDAP_BIND_TIMEOUT Statement" on page 136](#).

LDAP_FILTER

Specifies a filter to restrict the LDAP search for a PowerExchange user.

For more information, see ["LDAP_FILTER Statement" on page 137](#).

LDAP_HOST

Defines TCP/IP host details for an LDAP server that PowerExchange might use for user validation.

For more information, see ["LDAP_HOST Statement" on page 137](#).

LDAP_LOGIN_ATTRIBUTE

Specifies an LDAP attribute to use as the search key that is matched against PowerExchange user IDs.

For more information, see ["LDAP_LOGIN_ATTRIBUTE Statement" on page 138](#).

LDAP_OPENSSL

If you are using an OpenLDAP implementation, enables LDAP secure connections and specifies TLS certificate information for an LDAP TLS connection.

For more information, see ["LDAP_OPENSSL Statement" on page 138](#).

LDAP_PORT

Defines the TCP/IP port for an LDAP server that PowerExchange might use for user validation.

For more information, see ["LDAP_PORT Statement" on page 139](#).

LDAP_SASL_MECH

Specifies the Simple Authentication and Security Layer (SASL) authentication mechanism to be used with LDAP certificate-based security. The only valid mechanism is EXTERNAL.

For more information, see ["LDAP_SASL_MECH Statement" on page 140](#).

LDAP_SCOPE

Defines the depth of the search of the LDAP tree within the search base in LDAP searches.

For more information, see ["LDAP_SCOPE Statement" on page 140](#).

LDAP_SEARCH_TIMEOUT

Specifies the time, in seconds, that a PowerExchange Listener or PowerExchange Logger for Linux, UNIX, and Windows waits to receive a search result during validation of a PowerExchange user against LDAP.

For more information, see ["LDAP_SEARCH_TIMEOUT Statement" on page 141](#).

LDAP_TLS

If you are using an Oracle LDAP implementation, enables LDAP secure connections and specifies TLS certificate information for an LDAP TLS connection.

For more information, see ["LDAP_TLS Statement" on page 141](#).

SECURITY

Controls PowerExchange user authentication and access to resources and commands, including LDAP authentication, and specifies whether to use OpenLDAP or Oracle LDAP client libraries.

For more information, see ["SECURITY Statement" on page 171](#).

USE_DB_AUTH

When LDAP authentication is enabled, controls whether to use LDAP validation for relational connections. By default, database authentication is used for relational databases.

For more information, see ["USE_DB_AUTH Statement" on page 199](#).

Example DBMOVER LDAP Configurations

Review the example LDAP configurations to determine which DBMOVER statements you need to define.

If the SECURITY statement in an example does not include the optional fourth parameter, the example uses the Oracle LDAP client libraries, the default setting.

Example: Basic LDAP Configuration

The following example shows the DBMOVER statements for a basic LDAP configuration:

```
SECURITY=(1,N,LDAP)
LDAP_HOST=(myldapserver,389)
LDAP_BIND_DN="uid=hmiller,ou=People,dc=example,dc=com"
LDAP_BIND_PWD=hillock
LDAP_BASE=("ou=People,dc=example,dc=com")
```

This example directs PowerExchange user authentication to the LDAP server at host `myldapserver` on port 389. The search user binds with an LDAP identity of `uid=hmiller,ou=People,dc=example,dc=com` using password `hillock`. The LDAP search is based at `ou=People,dc=example,dc=com`.

Example: LDAP Configuration with a Filter and Login Attribute

The following example shows the DBMOVER statements for an LDAP configuration that defines a filter and login attribute:

```
SECURITY=(1,N,LDAP)
LDAP_HOST=(myldapserver,389)
LDAP_BIND_DN="uid=hmiller,ou=People,dc=example,dc=com"
LDAP_BIND_EPWD=47D95F9932FB5F67
LDAP_BASE=("ou=People,dc=example,dc=com")
LDAP_FILTER="(objectclass=person)"
LDAP_LOGIN_ATTRIBUTE=cn
```

This filter restricts the LDAP search to LDAP entries with an `objectclass` attribute that matches the value `person`.

Filtering can speed up searches because LDAP indexes can be used to reduce the number of searched entries. Without the LDAP_FILTER statement, the default filter of (objectclass=*) is used, which would match entries that have any objectclass value.

The login attribute of cn directs the search for an LDAP entry that has a common name that matches the PowerExchange user identity.

Note: In an actual application, because common name is not unique, you would not use it as a login key.

The LDAP_BIND_EPWD statement specifies an encrypted password for the search user.

Example: LDAP Configuration with Multiple Search Bases

The following example shows the DBMOVER statements for an LDAP configuration with multiple search bases:

```
SECURITY=(1,N,LDAP)
LDAP_HOST=(myldapserver,389)
LDAP_BIND_DN="uid=hmiller,ou=People,dc=example,dc=com"
LDAP_BIND_EPWD=47D95F9932FB5F67
LDAP_BASE=("ou=global,dc=example,dc=com")
LDAP_BASE=("ou=prod,dc=example,dc=com")
LDAP_BASE=("ou=dev,dc=example,dc=com")
LDAP_BASE=("ou=hr,dc=example,dc=com")
LDAP_BASE=("ou=sales,dc=example,dc=com")
```

In this example, the LDAP search proceeds through each base in the order listed until a match is found. Searching starts at ou=global,dc=example,dc=com and continues through subsequent bases if necessary.

Example: LDAP Configuration with Multiple Search Bases and a Filter

The following example shows the DBMOVER statements for an LDAP configuration with multiple search bases and a filter:

```
SECURITY=(1,N,LDAP)
LDAP_HOST=(myldapserver,389)
LDAP_BIND_DN="uid=hmiller,ou=People,dc=example,dc=com"
LDAP_BIND_EPWD=47D95F9932FB5F67
LDAP_BASE=("ou=global,dc=example,dc=com")
LDAP_BASE=("ou=prod,dc=example,dc=com")
LDAP_BASE=("ou=dev,dc=example,dc=com")
LDAP_BASE=("ou=hr,dc=example,dc=com")
LDAP_BASE=("ou=sales,dc=example,dc=com")
LDAP_FILTER="(|(initials=aa)(st=AK))"
```

This example uses a more complex filter that is the logical OR (|) of the conditions initials=aa and st=AK.

Example: LDAP Configuration with Failover

The following example shows the DBMOVER statements for an LDAP configuration with failover:

```
SECURITY=(1,N,LDAP)
LDAP_HOST=(myldapserver1,389)
LDAP_HOST=(myldapserver2,389)
LDAP_BIND_DN="uid=hmiller,ou=People,dc=example,dc=com"
LDAP_BIND_EPWD=47D95F9932FB4E56
LDAP_BASE=("ou=People,dc=example,dc=com")
```

LDAP failover is supported whenever the DBMOVER configuration file includes more than one LDAP_HOST statement. In this example, a search is first directed to myldapserver1. If that server is unavailable, the search is directed to myldapserver2. Subsequent searches are directed to myldapserver1 when it is available again.

Example: Level 1 LDAP Security - Secure LDAP Configuration with Encryption and Server Authentication

The following example shows the DBMOVER statements for a secured LDAP configuration with encryption and server authentication:

```
SECURITY=(1,N,LDAP)
LDAP_HOST=(myldapserver,636)
LDAP_BIND_DN="uid=hmiller,ou=People,dc=example,dc=com"
LDAP_BIND_EPWD=47D95F9932FB4E56
LDAP_BASE=("ou=People,dc=example,dc=com")
LDAP_TLS=(CAPATH=/pwx_truststore)
```

In this example, the LDAP_TLS statement enables encryption and configures Transport Level Security (TLS) to use the truststore directory /pwx_truststore. The truststore directory is an NSS certificate database that is configured to contain certificates that are sufficient to authenticate the LDAP server.

The LDAP_HOST statement directs the encrypted traffic to the secure port 636 of the LDAP server.

Example: Level 2 LDAP Security - Secure LDAP Configuration with Client Authentication

The following example shows DBMOVER statements for secure LDAP configuration with client authentication:

```
SECURITY=(1,N,LDAP)
LDAP_HOST=(myldapserver,636)
LDAP_BIND_DN="uid=hmiller,ou=People,dc=example,dc=com"
LDAP_BIND_EPWD=47D95F9932FB5F67
LDAP_BASE=("ou=People,dc=example,dc=com")
LDAP_TLS=(PASS=clientauth,
          KEYNAME=hmiller-cert,
          CAPATH=/pwx_store)
```

The /pwx_store directory is an NSS certificate database that is configured as a truststore and keystore. The database has entries for server and client authentication.

For server authentication, the truststore must contain certificates that are sufficient to authenticate the LDAP server.

For client authentication, the following entries must be present:

- A certificate (with alias name hmiller-cert, as indicated by the KEYNAME parameter) that identifies the client (PowerExchange) to the LDAP server. The LDAP server truststore must be configured with certificates that are sufficient to authenticate this client certificate.
- The private key associated with the hmiller-cert certificate. Accessing this key requires the password clientauth, as indicated by the PASS parameter.

The LDAP_HOST statement directs the encrypted traffic to the secure port 636 of the LDAP server.

Example: Level 3 LDAP Security - Secure LDAP Configuration with SASL External Authentication

The following example shows the DBMOVER statements for a secured LDAP configuration using the SASL EXTERNAL mechanism:

```
SECURITY=(1,N,LDAP)
LDAP_HOST=(myldapserver,636)
LDAP_BASE=("ou=People,dc=example,dc=com")
LDAP_TLS=(PASS=clientauth,
          KEYNAME=hmiller-cert,
          CAPATH=/pwx_store)
LDAP_SASL_MECH=EXTERNAL
```

This example uses the trustore and keystore directory, `pxw_store`, which is specified in the `CAPATH` parameter of the `LDAP_TLS` statement.

The `LDAP_SASL_MECH=EXTERNAL` statement indicates that the LDAP server will derive the search user credentials from the client certificate. To derive these credentials, the LDAP server must have the certificate mapping configured. For example, with Oracle Directory Server, the mapping is configured by the Oracle `certmap.conf` file.

Example: Level 1 LDAP Security - Secure LDAP Configuration with Encryption, Server Authentication, and StartTLS Extension

The following example extends Level 1 LDAP security to use the StartTLS extension:

```
SECURITY=(1,N,LDAP)
LDAP_HOST=(myldapserver,389)
LDAP_BIND_DN="uid=hmiller,ou=People,dc=example,dc=com"
LDAP_BIND_EPWD=47D95F9932FB4E56
LDAP_BASE=("ou=People,dc=example,dc=com")
LDAP_TLS=(CAPATH=/pxw_truststore,
          START_TLS=Y)
```

The `LDAP_HOST` statement directs traffic to plain text port 389 of the LDAP server. The `START_TLS=Y` parameter in the `LDAP_TLS` statement specifies that this traffic is encrypted.

Example: Level 2 LDAP Security - Secure LDAP Configuration with Client Authentication and StartTLS Extension

The following example extends Level 2 LDAP security to use the StartTLS extension:

```
SECURITY=(1,N,LDAP)
LDAP_HOST=(myldapserver,389)
LDAP_BIND_DN="uid=hmiller,ou=People,dc=example,dc=com"
LDAP_BIND_EPWD=47D95F9932FB5F67
LDAP_BASE=("ou=People,dc=example,dc=com")
LDAP_TLS=(PASS=clientauth,
          KEYNAME=hmiller-cert,
          CAPATH=/pxw_store,
          START_TLS=Y)
```

The `LDAP_HOST` statement directs network traffic to plain text port 389 of the LDAP server. The `START_TLS=Y` parameter in the `LDAP_TLS` statement specifies that this traffic is encrypted.

Example: Level 3 LDAP Security - Secure LDAP Configuration with SASL External Authentication and StartTLS Extension

The following example extends Level 3 LDAP security to use the StartTLS extension:

```
SECURITY=(1,N,LDAP)
LDAP_HOST=(myldapserver,389)
LDAP_BASE=("ou=People,dc=example,dc=com")
LDAP_TLS=(PASS=clientauth,
          KEYNAME=hmiller-cert,
          CAPATH=/pxw_store,
          START_TLS=Y)
LDAP_SASL_MECH=EXTERNAL
```

The `LDAP_HOST` statement directs traffic to plain text port 389 of the LDAP server. The `START_TLS=Y` parameter in the `LDAP_TLS` statement specifies that this traffic is encrypted.

Example: Level 1 LDAP Security - Secure OpenLDAP Configuration with Encryption and Server Authentication

The following example shows the DBMOVER statements for a secured OpenLDAP configuration with encryption and server authentication:

```
SECURITY=(1,N,LDAP,OPEN_LDAP)
LDAP_HOST=(myldapserver,636)
LDAP_BIND_DN="uid=hmiller,ou=People,dc=example,dc=com"
LDAP_BIND_EPWD=47D95F9932FB4E56
LDAP_BASE=("ou=People,dc=example,dc=com")
LDAP_OPENSSL=(CAPATH=/pwx_truststore)
```

In this example, the fourth parameter of the SECURITY statement specifies to load OpenLDAP client libraries. The LDAP_OPENSSL statement enables encryption and configures Transport Level Security (TLS) to use the truststore directory /pwx_truststore. The truststore directory holds Certificate Authority certificate files, in PEM format, that allow PowerExchange to validate a signed certificate returned by the LDAP server.

The LDAP_HOST statement directs the encrypted traffic to the secure port 636 of the LDAP server.

Example: Level 2 LDAP Security - Secure OpenLDAP Configuration with Client Authentication

The following example shows DBMOVER statements for secure OpenLDAP configuration with client authentication:

```
SECURITY=(1,N,LDAP,OPEN_LDAP)
LDAP_HOST=(myldapserver,636)
LDAP_BIND_DN="uid=hmiller,ou=People,dc=example,dc=com"
LDAP_BIND_EPWD=47D95F9932FB5F67
LDAP_BASE=("ou=People,dc=example,dc=com")
LDAP_OPENSSL=(CAPATH=/pwx_store,
               CERTFILE=/pwx_store/mycert.pem,KEYFILE=/pwx_store/mykey.pem,
               PASS=clientauth
            )
```

The /pwx_store directory serves as a truststore and keystore. The directory has entries for server and client authentication.

For server authentication, the truststore must contain Certificate Authority certificates, in PEM format, that are sufficient to validate the LDAP server.

For client authentication, the following entries must be present:

- The certificate file and key file that identify the client (PowerExchange) to the LDAP server. The LDAP server truststore must be configured with Certificate Authority certificates, in PEM format, that are sufficient to validate this client certificate.
- The private key associated with the client certificate. The key file is encrypted. Accessing this key requires the password `clientauth`, as indicated by the PASS parameter.

The LDAP_HOST statement directs the encrypted traffic to the secure port 636 of the LDAP server.

Example: Level 3 LDAP Security - Secure OpenLDAP Configuration with SASL External Authentication

The following example shows the DBMOVER statements for a secured LDAP configuration using the SASL EXTERNAL mechanism:

```
SECURITY=(1,N,LDAP,OPEN_LDAP)
LDAP_HOST=(myldapserver,636)
LDAP_BASE=("ou=People,dc=example,dc=com")
LDAP_OPENSSL=(CAPATH=/pwx_store,
               CERTFILE=/pwx_store/mycert.pem,KEYFILE=/pwx_store/mykey.pem,
```

```
PASS=clientauth
)
LDAP_SASL_MECH=EXTERNAL
```

This example uses the trustore and keystore directory, `pw_x_store`, which is specified in the `CAPATH` parameter of the `LDAP_OPENSSL` statement.

The `LDAP_SASL_MECH=EXTERNAL` statement indicates that the LDAP server will derive the search user credentials from the client certificate. To derive these credentials, the LDAP server must have the certificate mapping configured.

PowerExchange Selective Sign-on

You can use PowerExchange selective sign-on to control connection access to PowerExchange. With selective sign-on, PowerExchange checks the PowerExchange sign-on file to verify access for the supplied user ID and, optionally, the IP address. PowerExchange performs selective sign-on checking when a PowerExchange Listener accepts a new connection over TCP/IP, and prior to starting a subtask for the request.

Tip: On z/OS and i5/OS, PowerExchange can also utilize security facilities of the z/OS and i5/OS operating systems to authenticate users. Use PowerExchange selective sign-on in conjunction with the user authentication option of the `SECURITY` statement. Selective sign-on does not perform password validation and provides only a minimal level of security.

Configuring PowerExchange Selective Sign-on

Configure PowerExchange selective sign-on by completing the following steps.

1. In the `SECURITY` statement in the `DBMOVER` configuration file, set the second parameter to `Y`.
Note: The first parameter of the `SECURITY` statement only controls whether PowerExchange uses the sign-on file to authorize users to issue `infacmd pwx` commands to a PowerExchange application service in the Informatica domain or `pwxcmd` commands to a PowerExchange process. The first parameter does not control connections to PowerExchange or access to PowerExchange Listener `LISTTASK` and `STOPTASK` commands from the PowerExchange Navigator.
2. Set up a PowerExchange sign-on file on the Linux, UNIX, or Windows system where the PowerExchange Service or the PowerExchange process runs.
In the file, specify the allowed users and optionally the IP addresses from which those users are allowed access.
3. Start or restart the PowerExchange Listener.

RELATED TOPICS:

- [“PowerExchange Sign-on File” on page 294](#)

PowerExchange Sign-on File

Use the PowerExchange sign-on file to authorize users to connect to PowerExchange and to issue commands.

The PowerExchange sign-on file authorizes users to complete the following tasks:

- Connect to PowerExchange systems from specific IP addresses.
- Issue specific `pwxcmd` and `infacmd pwx` commands. You issue `infacmd pwx` commands to a PowerExchange Service and `pwxcmd` commands to PowerExchange processes that are not managed by an application service. You can issue commands to a PowerExchange Service or process that runs on any Linux, UNIX, or Windows system.
- On Windows, issue PowerExchange Listener commands to a PowerExchange Listener through the PowerExchange Navigator.

PowerExchange Sign-on File Syntax

Use the following syntax for the sign-on file statements:

```
[AUTHGROUP=(group_id,COMMANDS=(command1,command2,...))]  
USER=(user_ID[@domain]  
    ALLOW={F|N|Y}  
    [AUTHGROUP=group_id]  
    [IP=nnnn.nnnn.nnnn.nnnn]  
    [TASKCNTRL={D|S}]  
)
```

The following rules and guidelines apply:

- Maximum statement length is 64 characters.
- Statements can begin in any column.
- Statements cannot be continued and must be coded on a single line.
- You can specify one or more AUTHGROUP or USER statements in a sign-on file.
- When PowerExchange encounters a USER statement that matches all or part of the user ID, PowerExchange stops reading the sign-on file.
- Comment lines must begin with `/*`, and must be on a separate line from any statements.
- On a Linux, UNIX, or Windows system, if you do not create specific authorization groups to authorize users to issue `pwxcmd` and `infacmd pwx` commands, you must include the following statements in the sign-on file on all target Linux, UNIX, or Windows systems to which users issue commands:

```
AUTHGROUP=(ANY,COMMAND=(*))  
USER=(*,ALLOW=Y,AUTHGROUP=ANY)
```

PowerExchange provides a sample sign-on file, in the required location, that you can customize.

The following table describes, by operating system, the name and required location of sign-on file:

Operating System	SIGNON File Name and Location
i5/OS	SIGNON member in the <i>datalib</i> /CFG file
Linux, UNIX, and Windows	signon.txt file in the PowerExchange installation directory
z/OS	SIGNON member in RUNLIB to which the DTLSGN DD in the PowerExchange Listener JCL points

PowerExchange Sign-on File Statements

In the sign-on file, define statements to authorize selective sign-on, PowerExchange Listener commands through PowerExchange Navigator, infacmd pwx commands, or pwxcmd commands.

Define the following statements:

- For selective sign-on and PowerExchange Listener command authorization, use the USER statement.
- For pwxcmd and infacmd pwx command authorization on a Linux, UNIX, or Windows system, use the AUTHGROUP and USER statements.

AUTHGROUP Statement

On Linux, UNIX, and Windows, limits access to specified pwxcmd and infacmd pwx commands to the users in the authorization group.

The AUTHGROUP statement has the following syntax:

```
AUTHGROUP=(group_id,COMMANDS=(command1,command2,...))
```

The AUTHGROUP statement has the following parameters:

group_id

Required. Specifies the name of the authorization group that has access to the specified pwxcmd and infacmd pwx commands. This name must match the authorization group name in a corresponding USER statement, which associates one or more user IDs with the authorization group.

COMMANDS=(command1,command2,...)

Required. Specifies one or more pwxcmd and infacmd pwx commands that users in this authorization group are authorized to run.

To authorize users to run all pwxcmd and infacmd pwx commands, specify an asterisk (*) instead of commands.

The following table describes the commands that you can specify for the PowerExchange Listener:

Command	Corresponds to infacmd pwx Command	Corresponds to pwxcmd Command
CLOSE	CloseListener	close
CLOSEFORCE	CloseForceListener	closeforce

Command	Corresponds to infacmd pwx Command	Corresponds to pwxcmd Command
LISTTASK	ListTaskListener	listtask
STOPTASK	StopTaskListener	stoptask

The following table describes the commands that you can specify for PowerExchange Condense:

Command	Corresponds to infacmd pwx Commands	Corresponds to pwxcmd Command
CONDENSE	-	condense
DISPLAYSTATUS	-	displaystatus
FILESWITCH	-	fileswitch
SHUTCOND	-	shutcond
SHUTDOWN	-	shutdown

The following table describes the commands that you can specify for the PowerExchange Logger for Linux, UNIX, and Windows:

Command	Corresponds to infacmd pwx Command	Corresponds to pwxcmd Command
CONDENSE	CondenseLogger	condense
DISPLAYALL	DisplayAllLogger	displayall
DISPLAYCHECKPOINTS	DisplayCheckpointsLogger	displaycheckpoints
DISPLAYCPU	DisplayCPULogger	displaycpu
DISPLAYEVENTS	DisplayEventsLogger	displayevents
DISPLAYMEMORY	DisplayMemoryLogger	displaymemory
DISPLAYRECORDS	DisplayRecordsLogger	displayrecords
DISPLAYSTATUS	DisplayStatusLogger	displaystatus
FILESWITCH	FileSwitchLogger	fileswitch
SHUTCOND	-	shutcond
SHUTDOWN	ShutDownLogger	shutdown

You do not need authorization to run the pwxcmd help and version commands.

Usage Notes:

On Linux, UNIX, and Windows, if you set the first parameter to 2 in the SECURITY statement and you do not include specific AUTHGROUP and USER statements in the sign-on files to control access to pwxcmd and infacmd pwx commands, you must include default AUTHGROUP and USER statements. In the sign-on files on

all Linux, UNIX, and Windows systems to which users issue pwxcmd and infacmd pwx commands, include the following default statements:

```
AUTHGROUP=(ANY, COMMAND=(*))  
USER=(*, ALLOW=Y, AUTHGROUP=ANY)
```

USER Statement

Use the USER statement to authorize the specified user and, optionally, IP address, to connect to PowerExchange.

Optionally, on Linux, UNIX, and Windows, use the USER statement to authorize the specified user to issue pwxcmd and infacmd pwx commands. On Windows, use the USER statement authorize users to issue PowerExchange Listener commands through the PowerExchange Navigator. The maximum statement length is 64 characters.

The USER statement has the following syntax:

```
USER=(user_ID[@domain]  
      ALLOW={F|N|Y}  
      [AUTHGROUP=group_id]  
      [IP=nnnn.nnnn.nnnn.nnnn]  
      [TASKCNTRL={D|S}]  
      )
```

The USER statement has the following parameters:

user_id[@domain]

Required. Specifies the user ID of a user. Optionally, on Windows, you can specify a domain name. User ID checking is case-sensitive.

You can use the following wildcard characters in the user ID:

- Asterisk (*), which represents 0 or more characters.
- Question mark (?), which represents exactly one character.

If the user ID contains an asterisk (*) or question mark (?) character as a part of the ID, precede these characters with the tilde (~) escape character.

The user specifications on a pwxcmd command and the USER statement in the sign-on file must match. If you specify a domain as part of a user ID in the sign-on file, you must also specify the domain in the user ID when you issue an infacmd pwx or pwxcmd command.

If you do not specify a domain as part of a user ID in a infacmd pwx or pwxcmd command, PowerExchange assumes that the user ID is local or a member of the same domain as the user ID under which the PowerExchange Listener runs. If you run the PowerExchange Listener as a Windows service and that service runs under the local user, you must specify the domain as part of a user ID in an infacmd pwx or pwxcmd command.

ALLOW={F|N|Y}

Required. Specifies one of the following values:

- **F.** Indicates that additional ALLOW parameters follow.
- **N.** The user cannot connect on to the PowerExchange. If an IP address is specified, the user is restricted from that IP address only.
- **Y.** The user can connect to PowerExchange from any IP address, unless a specific IP address is specified in the IP parameter.

AUTHGROUP=group_ID

On Linux, UNIX, or Windows, specifies the name of the authorization group to which this user is assigned. This name must match the authorization group name specified in a corresponding AUTHGROUP statement, which authorizes the users in the authorization group to issue one or more pwxcmd and infacmd pwx commands.

IP=nnnn.nnnn.nnnn.nnnn

Optional. Specifies an IP address of a user. The *nnnn* value of an IP address, called an octet, is a number from 0 through 255.

If the last octet of the IP address is zero, then PowerExchange uses the octet values to cover number from 0 to 255. For example, the following IP parameter specification covers all IP addresses from 10.10.10.0 to 10.10.10.255:

```
IP=10.10.10.0
```

If an octet begins with zero, then PowerExchange assumes the value specified is octal. For example, if you specify 0120, PowerExchange converts this value to 80. Generally, you should specify decimal numbers for the octet values.

TASKCNTRL={D|S}

Optional. On Windows, authorizes the user to issue the LISTTASK or STOPTASK command from the PowerExchange Navigator. Valid values are:

- **D**. The user can issue the LISTTASK command from the PowerExchange Navigator.
- **S**. The user can issue the STOPTASK command from the PowerExchange Navigator.

This option is valid only if SECURITY=(1,Y) in the DBMOVER configuration file.

To issue a command from the PowerExchange Navigator, click **TASK_CNTRL** in the **DB Type** list in the **Database Row Test** dialog box.

PowerExchange Sign-on File Examples

The following examples demonstrate how to define the PowerExchange sign-on file to control connection to PowerExchange and access to pwxcmd and infacmd pwx commands.

PowerExchange Selective Sign-on Examples

The following example USER statements in a sign-on file show how to control connection to PowerExchange based on user ID and IP address.

To enable all users to connect to PowerExchange, code the following USER statement in the sign-on file:

```
USER= (* ALLOW=Y)
```

To restrict access to PowerExchange to users with user IDs that begin with SYS, code the following USER statements in the sign-on file:

```
USER= (SYS* ALLOW=Y)
USER= (* ALLOW=N)
```

To allow users to sign on from multiple IP addresses, code multiple USER statements with the ALLOW=F parameter. When PowerExchange encounters the first USER statement with ALLOW=Y or N and a user ID that matches, PowerExchange stops reading the sign-on file.

The following statements enable USER1 to connect to PowerExchange from three different IP addresses:

```
USER= (USER1 ALLOW=F IP=10.7.16.25)
USER= (USER1 ALLOW=F IP=10.7.16.26)
USER= (USER1 ALLOW=Y IP=10.7.16.30)
```

In the following statements, the ALLOW=Y parameter in the second and third USER statements enable USER1 to sign on from IP addresses 10.7.16.25 and 10.7.16.26 only:

```
USER=(USER1 ALLOW=F IP=10.7.16.25)
USER=(USER1 ALLOW=Y IP=10.7.16.26)
USER=(USER1 ALLOW=Y IP=10.7.16.30)
```

pwxcmd and infacmd pwx Command Authorization Examples

On Linux, UNIX, and Windows, define AUTHGROUP and USER statements in the sign-on file to authorize users to run specific pwxcmd and infacmd pwx commands.

Define statements as follows:

- The AUTHGROUP statement authorizes users in an authorization group to issue one or more pwxcmd and infacmd pwx commands.
- The USER statement associates one or more user IDs with an authorization group. Also, the ALLOW parameter in a USER statement specifies whether users are allowed to connect to PowerExchange.

The following statements assign user ops99 to the ADMIN authorization group, enable ops99 to connect to PowerExchange, and authorize ops99 to run all pwxcmd and infacmd pwx commands:

```
AUTHGROUP=(ADMIN,COMMANDS=(*))
USER=(ops99 ALLOW=Y AUTHGROUP=ADMIN)
```

The following statements assign users with user IDs that begin with SYS to the OPER authorization group, enable them to connect to PowerExchange, and authorize them to run specified pwxcmd and infacmd pwx commands:

```
AUTHGROUP=(OPER,COMMANDS=(close,closeforce,listtask,stoptask))
USER=(SYS* ALLOW=Y AUTHGROUP=OPER)
```

The following statements enable all users to connect to PowerExchange and to run the infacmd pwx ListTaskListener and pwxcmd listtask command:

```
AUTHGROUP=(BASIC,COMMANDS=(LISTTASK))
USER=(* ALLOW=Y AUTHGROUP=BASIC)
```

CHAPTER 7

Secure Sockets Layer Support

This chapter includes the following topics:

- [SSL Security on a PowerExchange Network, 300](#)
- [PowerExchange SSL Architecture, 301](#)
- [FIPS 140-2 Compliance, 301](#)
- [PWXUGSK Utility, 303](#)
- [PWXUSSL Utility, 303](#)
- [PowerExchange SSL Configuration Steps, 304](#)
- [SSL Server Configuration on z/OS, 305](#)
- [SSL Client and Server Configuration on Linux, UNIX, and Windows, 309](#)
- [Verifying Connections, 319](#)

SSL Security on a PowerExchange Network

You can configure Secure Sockets Layer (SSL) communication on a PowerExchange network to ensure secure communication. To configure SSL communication, establish certificates and keys that authorize the secure connection between systems and enable encryption and decryption of data.

Each server or client machine has SSL private key and SSL certificate components. You can activate these components by configuring PowerExchange.

After configuration, the SSL handshake and acceptance set up the secure connection. The individual data messages are encrypted using the session key that is encoded and exchanged during the handshake.

PowerExchange supports SSL communication for the following operating systems:

- Linux
- UNIX
- Windows
- z/OS

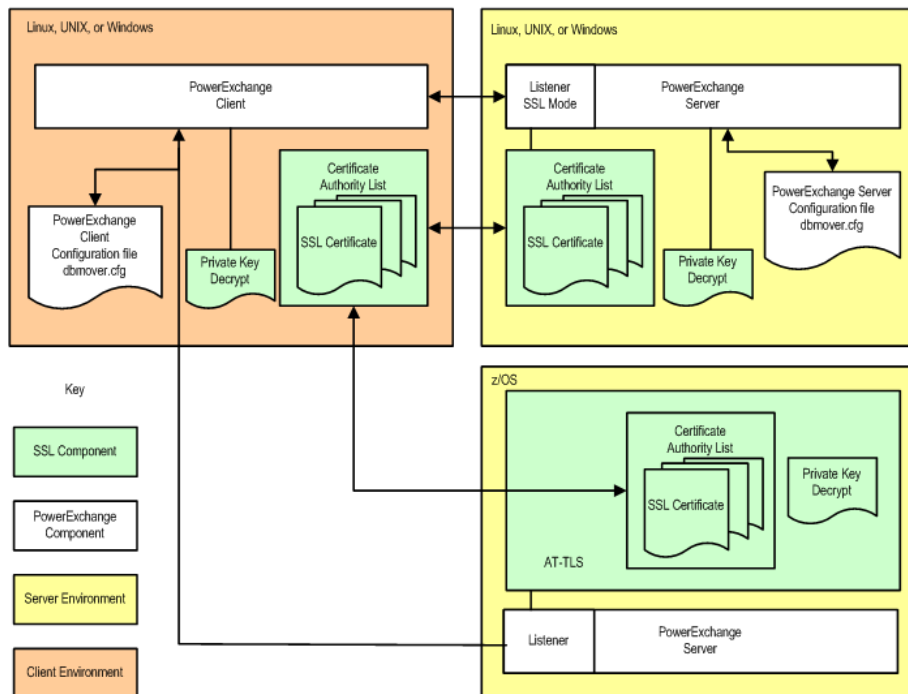
Note: Unless otherwise noted, the term *SSL* is used to denote both the Secure Sockets Layer (SSL) and Transport Layer Security (TLS) protocols.

PowerExchange SSL Architecture

The PowerExchange SSL architecture includes the following components:

- OpenSSL installed on each SSL client and server on Linux, UNIX, or Windows
- PowerExchange SSL configured on each SSL client and server on Linux, UNIX, or Windows
- AT-TLS installed and configured for PowerExchange on each SSL server on z/OS
- PowerExchange Listener configured on z/OS
- X.509 certificates installed on each SSL client and server

The following figure illustrates the PowerExchange SSL architecture:



You can use SSL communication for some, all, or none of the connections on a PowerExchange network.

For example, you might configure connections as follows:

- Configure PowerExchange Listeners to use separate ports for SSL and non-SSL connections.
- Configure the PowerCenter Integration Service client to use SSL connections to PowerExchange Listeners.
- Configure the PowerExchange Navigator and PowerCenter clients to use non-SSL connections to PowerExchange Listeners.

FIPS 140-2 Compliance

SSL, in an appropriate environment, complies with the Federal Information Processing Standard (FIPS) Publication 140-2. By configuring PowerExchange appropriately, creating the necessary certificates, and selecting appropriate algorithms, you can achieve FIPS 140-2 Security Level 1 compliance on a

PowerExchange network. To ensure that your PowerExchange network meets the requirements for FIPS 140-2 compliance, consult your security administrator.

FIPS 140-2 Compliant Cipher Suites

During an SSL handshake, the client and server agree on a symmetric algorithm to use to encrypt data during the session. The client offers a list of cipher suites, and the server selects one from the list. For the PowerExchange network to be FIPS 140-2 compliant, the selected cipher suite must be FIPS 140-2 compliant.

On Linux, UNIX, or Windows clients or servers, PowerExchange uses the OpenSSL runtime engine. When a client and server are both using OpenSSL, the cipher suite that PowerExchange selects is FIPS 140-2 compliant.

On z/OS, AT-TLS manages SSL sessions. The order of cipher suites in the TTLS cipherParms statement in the AT-TLS policy file is important. The server selects the first cipher suite in the list that matches one offered by the client. In this process, ciphers are identified using hexadecimal cipher suite numbers.

To ensure that a z/OS server selects a FIPS 140-2 compliant cipher suite, verify that the first cipher suite in the TTLS cipherParms list matches one of the FIPS 140-2 compliant cipher suites that OpenSSL supports.

The following table is a partial list of FIPS 140-2 compliant cipher suites that OpenSSL and AT-TLS both support:

OpenSSL Cipher Suite Name	AT-TLS Cipher Suite Name	Hexadecimal Value
DHE-RSA-AES256-SHA	TLS_DHE_RSA_WITH_AES_256_CBC_SHA	39
DHE-DSS-AES256-SHA	TLS_DHE_DSS_WITH_AES_256_CBC_SHA	38
AES256-SHA	TLS_RSA_WITH_AES_256_CBC_SHA	35
EDH-RSA-DES-CBC3-SHA	TLS_DHE_RSA_WITH_3DES_EDE_CBC_SHA	16
EDH-DSS-DES-CBC3-SHA	TLS_DHE_DSS_WITH_3DES_EDE_CBC_SHA	13
DES-CBC3-SHA	TLS_RSA_WITH_3DES_EDE_CBC_SHA	0A
DHE-RSA-AES128-SHA	TLS_DHE_RSA_WITH_AES_128_CBC_SHA	33
DHE-DSS-AES128-SHA	TLS_DHE_DSS_WITH_AES_128_CBC_SHA	32
AES128-SHA	TLS_RSA_WITH_AES_128_CBC_SHA	2F

For a complete list of algorithms that AT-TLS supports, see the AT-TLS documentation. For a complete list of algorithms that OpenSSL supports on your machine, run the PWXUSSL utility on Linux, Unix, and Windows. For a complete list of algorithms supported by your z/OS system, run the PWXUGSK utility. For more information about PWXUSSL and PWXUGSK, see the *PowerExchange Utilities Guide*.

Note: The optional, no-charge CPACF feature available on IBM System z machines provides machine instructions to accelerate hashing algorithms and symmetric key encryption and decryption used with SSL. For performance reasons, you may wish to use only those cryptographic suites for which hardware assists are available. For example, CPACF supports AES-128 on z9 and later machines and AES-256 on z10 and z196 machines.

FIPS 140-2 Compliance Considerations on z/OS

FIPS 140-2 level 1 support in z/OS System SSL requires z/OS 1.10 with the fixes for APAR OA26457, or z/OS 1.11 or later. Additional fixes and z/OS configuration changes are also required.

In particular, the operating system must include the Cryptographic Services Security Level 3 (FMID JCPT391) component of z/OS System SSL.

z/OS AT-TLS uses z/OS System SSL. For z/OS 1.12 and later, AT-TLS provides the following features in support of FIPS 140-2 compliance:

- A configuration parameter to request that System SSL uses only FIPS 140-2 compliant encryption methods
- Symbolic names for recent encryption methods such as those using AES-256 bit encryption

For earlier version of z/OS, you can enforce the use of FIPS-compliant encryption by specifying the candidate encryption methods as hexadecimal codes rather than symbolic names.

For more information about achieving FIPS 140-2 Level 1 compliance with System SSL, see the following IBM publications:

- *Cryptographic Services System Secure Sockets Layer Programming*
- *APAR OA26457 System Secure Sockets Layer Programming*

PWXUGSK Utility

Use the PowerExchange PWXUGSK utility to generate reports about SSL libraries and certificates on z/OS.

The PWXUGSK utility supports the following commands and reports:

- PING command. Verifies that a secure connection can be established between the machine from which you issue the command and a PowerExchange Listener on a remote node.
- Certificates report. Reports certificate information from a RACF keyring or SAF database.
- Ciphers report. Reports the cipher suites that are available in the cryptographic library to a specific user.
- Error codes report. Reports error codes generated during an attempt to establish a secure connection. This report can be used to help diagnose connection errors.

For more information about PWXUGSK, see the *PowerExchange Utilities Guide*.

PWXUSSL Utility

Use the PowerExchange PWXUSSL utility to generate reports about SSL libraries and certificates on Linux, UNIX, and Windows.

The PWXUSSL utility supports the following commands and reports:

- CONVERT_CERT_PKCS12_PEM command. Converts certificates that were created on z/OS in PKCS12DER format to the PEM format that can be used on Linux, Unix, and Windows machines.
- PING command. Verifies that a secure connection can be established between the machine from which you issue the command and a PowerExchange Listener on a remote node.

- Certificate report. Reports information from a certificate chain file.
- Ciphers report. Reports the cipher suites that are available in the OpenSSL cryptographic library.
- Codes report. Reports all available OpenSSL return codes when verification of peer certificates fails.
- Configuration report. Reports information about the DBMOVER configuration file settings that are in effect when the report is run.
- Error codes report. Reports error codes returned from SSL processing during an attempt to establish a secure connection between the PowerExchange Listener and client. This report can be used to help diagnose connection errors.
- Version report. Reports the version of OpenSSL that was used to build the cryptographic library.

For more information about PWXUSL, see the *PowerExchange Utilities Guide*.

PowerExchange SSL Configuration Steps

Before you begin SSL configuration for PowerExchange, your organization should have a local CA certificate from a well-known CA vendor. A self-signed CA certificate can be generated instead for internal use, such as connections within your organization's network or internal testing.

The steps described in this task should be performed by security administrators. Security administrators have specific permissions and system access that allow them to generate and manage security certificates and policy files.

Note: All certificates created for use with PowerExchange must be generated to the X.509 standard. For example, the PKCS7 format meets the X.509 standard, so it can be used to generate the certificates.

To implement SSL support in PowerExchange, complete the following tasks:

1. Configure each z/OS server.
 - a. Update the AT-TLS policy file.
 - b. Create a personal certificate.
 - c. Configure the PowerExchange Listener in the DBMOVER file.
2. Configure each Linux, UNIX, or Windows server.
 - a. Create a CA certificate.
 - b. Create a personal certificate.
 - c. Customize the DBMOVER configuration file on the server.
3. If your organization requires client validation, configure each Linux, Unix, or Windows client:
 - a. Create a CA certificate for the client.
 - b. Create a personal certificate for the client.
 - c. Customize the DBMOVER configuration file to use client validation.
4. Make the certificates available to servers and clients that require authentication.
5. Verify the secure connections between PowerExchange clients and servers.

SSL Server Configuration on z/OS

To configure an SSL server on z/OS, perform the following tasks:

- Update the AT-TLS policy file.
- Create a personal certificate.
- Configure the PowerExchange Listener.

Note: Because AT-TLS is part of the TCP/IP stack, the network system programmer typically holds the responsibility for maintaining the AT-TLS policy file and infrastructure.

Updating the AT-TLS Policy File

On z/OS release 1.7 and later, Application Transparent - Transport Layer Security (AT-TLS) uses a Communications Server policy file to determine which sessions will use the SSL protocol.

The policy file contains statements that define the SSL connections for the z/OS jobs that use SSL. Any job that communicates with an SSL Listener must have a statement in the policy file to define the secure connection. You can add multiple job names to the AT-TLS policy file.

To configure AT-TLS for PowerExchange, verify that the AT-TLS infrastructure is set up, and add a rule to the policy file.

For more information about AT-TLS, see the IBM AT-TLS documentation.

Setting Up the AT-TLS Infrastructure

If the AT-TLS infrastructure is not already set up, you must set it up before you customize it for PowerExchange SSL.

1. Activate the Communications Server Policy Agent.
2. Create the AT-TLS policy file for PowerExchange and make it available to the Policy Agent.

The IBM Configuration Assistant for z/OS Communications Server assists in building an AT-TLS configuration and policy files. This is a graphical user interface that is installed with the IBM z/OS Management Facility.

Get SSL certificates for all machines that use SSL connection in the PowerExchange environment. The system administrator or security administrator typically supplies these certificates.

RELATED TOPICS:

- [“Example AT-TLS Policy File for PowerExchange” on page 307](#)

Adding an AT-TLS Rule

To add a rule, edit the policy file or use the IBM Configuration Assistant for z/OS Communications Server.

The IBM Configuration Assistant for z/OS Communications Server is installed with the IBM z/OS Management Facility.

The following table shows the statements to include when you add a rule:

Statement	Value
LocalPortRange	PowerExchange Listener port number.
Jobname	PowerExchange Listener job name.
Direction	Direction of communication. Specify Inbound to indicate that communication proceeds from client to Listener.
TTLSTGroupActionRef	References an existing <i>group_action</i> that is defined in another section of the policy file.
TTLSEnvironmentActionRef <i>environment_action</i>	References an existing <i>environment_action</i> that is defined in another section of the policy file.

Example Rule

The following statements show an example rule:

```
TTLSTRULE JOB_JBBV861
{
  LocalPortRange      13132
  Jobname              JBBV861
  Direction            Inbound
  TTLSTGroupActionRef gActEnableTTLST
  TTLSEnvironmentActionRef eActServerDefault
}
```

References to Existing Sections of Policy File

The following table describes the existing sections of the policy file that the rule references:

Statement	Sub-Statement	Value
TTLST Group Action	TTLSEnabled	On
TTLST Group Action	CtraceClearText	Off
TTLST Group Action	Trace	7
TTLSEnvironmentAction	HandshakeRole	For servers, specifies one of the following values: - Server . The Listener acts as the SSL server and does not require client authentication. - ServerWithClientAuth . The Listener acts as the SSL server and requires client authentication.
TTLSEnvironmentAction	TTLSCipherParmsRef	References the TTLSCipherParms statement.
TTLSEnvironmentAction	TTLSTKeyRingParmsRef	References the TTLSTKeyRingParms statement.
TTLSTCipherParms	V3CipherSuites	Supported symmetric cipher suites.
TTLSTKeyRingParms	Keyring	Key ring that contains the personal and CA certificates.

The following statements show example sections of a referenced policy file:

```
TTLSTGroupAction  gActEnableTTLST
{
```

```

TTLSEnabled          On
CtracedClearText     Off
Trace                7
}
TTLSEnvironmentAction eActServerDefault
{
  HandshakeRole       Server
  TTLSCipherParmsRef  cipher1~AT-TLS__Silver
  TTLSKeyringParmsRef kATTLskeyring
}
TTLSCipherParms      cipher1~AT-TLS__Silver
{
  V3CipherSuites      TLS_RSA_WITH_DES_CBC_SHA
  V3CipherSuites      TLS_RSA_WITH_3DES_EDE_CBC_SHA
  V3CipherSuites      TLS_RSA_WITH_AES_128_CBC_SHA
}
TTLSTKeyRingParms    kATTLskeyring
{
  Keyring             ATTLs_keyring
}

```

Example AT-TLS Policy File for PowerExchange

The following example shows a complete AT-TLS policy file for PowerExchange:

```

##
## AT-TLS Policy Agent Configuration file for:
##   Image: MHZ2
##   Stack: TCPIP
## This is a sample AT TLS Policy Agent Configuration file for Power Exchange, based
## on a subset of a file originally created by the IBM z/OS Network Security
## Configuration Assistant application.
##
## It defines a listener (jobname PWXSSL85) which acts as SSL server
## (Direction=inbound) and a second listener (PWXSSLMX) which acts as SSL server only
## when connected via its port 18501.
##
## TLSRule 0~1 defines listener with jobname PWXSSL85 which as as SSL server for all
## connections (connection direction is inbound - from client to listener).
##   - IpAddrSet addr1 specifies that the rule applies to any IP address and port.
##   - Group Action gAct1 specifies that SSL is to be enabled for this connection.
##   - Environment Action eAct1 specifies that the listener acts as SSL server, with
##     keyring defined by the keyR-MHZ2 TTLSkeyRingParms statement.
##   - Connection Action cAct1 specifies that encryption parameters defined by
##     TTLSCipherParms statement cipher1~AT-TLS__Silver, and that the listener is to act
##     as SSL server (without client certificates).
##
## TLSRule 0~5 defines a similar listener with jobname PWXSSLMX which has more than one
## listener port. It is to use SSL only if communicating via its port 18501 (portrange
## statement portR1.
##
## TTLSCipherParms cipher1~AT-TLS__Silver shows a list on Cipher Suites which can be used
## for the connections.
##
TTLSTRule              0~1
{
  LocalAddrSetRef      addr1
  RemoteAddrSetRef     addr1
  Jobname               PWXSSL85
  Direction             Inbound
  Priority              255
  TTLSTGroupActionRef  gAct1
  TTLSEnvironmentActionRef eAct1
  TTLSConnectionActionRef cAct1
}
TTLSTRule              0~5~
{
  LocalAddrSetRef      addr1
  RemoteAddrSetRef     addr1
  LocalPortRangeRef    portR1
}

```

```

        Jobname                PWXSSLMX
        Direction              Inbound
        Priority                251
        TLSGroupActionRef      gAct1
        TLSEnvironmentActionRef eAct1
        TLSConnectionActionRef cAct1
    }
    TLSGroupAction              gAct1
    {
        TLSEnabled              On
        Trace                    7
    }
    TLSEnvironmentAction        eAct1
    {
        HandshakeRole            Server
        EnvironmentUserInstance  0
        TLSKeyringParmsRef       keyR~MHZ2
    }
    TLSEnvironmentAction        eAct2
    {
        HandshakeRole            Client
        EnvironmentUserInstance  0
        TLSKeyringParmsRef       keyR~MHZ2
    }
    TLSConnectionAction         cAct1
    {
        HandshakeRole            Server
        TLSCipherParmsRef         cipher1~AT-TLS__Silver
        Trace                      7
    }
    TLSConnectionAction         cAct2
    {
        HandshakeRole            Client
        TLSCipherParmsRef         cipher1~AT-TLS__Silver
        Trace                      7
    }
    TLSKeyringParms             keyR~MHZ2
    {
        Keyring                   ATTLS_keyring
    }
    TLSCipherParms              cipher1~AT-TLS__Silver
    {
        V3CipherSuites            TLS_RSA_WITH_DES_CBC_SHA
        V3CipherSuites            TLS_RSA_WITH_3DES_EDE_CBC_SHA
    }
    IpAddrSet                   addr1
    {
        Prefix                     0.0.0.0/0
    }
    PortRange                   portR1
    {
        Port                       18501
    }
}

```

Creating a Personal Certificate on z/OS

To create a personal or subject identification certificate use RACF or another facility such as ACF2 or TopSecret.

The following steps assume you are using RACF and that the RACF administrator has already created a suitable CA (site) certificate.

1. Create a key ring.
2. Create a personal certificate.
3. Connect the personal certificate to the key ring.
4. Connect the CA certificate to the key ring.

The following example illustrates these steps:

```
/* Create a Keyring for the application */
RACDCERT ID(MYUSERID) ADDRING(ATTLS_keyring)
SETROPTS RACLIST(DIGTCERT,DIGTNMAP) REFRESH

/* Create a certificate for the Server application */
RACDCERT ID(MYUSERID) GENCERT -
  SUBJECTSDN ( -
    O('MyCompany') -
    CN('MYUSERID.mymachine.myorganization.com') -
    OU('myorganizationunit') -
    C('GB') -
  )
  WITHLABEL('MYUSERIDCert1')-
  SIGNWITH(CERTAUTH LABEL('LOCALCA'))
SETROPTS RACLIST(DIGTCERT,DIGTNMAP) REFRESH

/* Connect the server certificate to the server's keyring.*/
RACDCERT ID(MYUSERID) CONNECT(ID(MYUSERID) - LABEL('MYUSERIDCert1') -
RING(ATTLS_keyring) -
DEFAULT -
USAGE(personal))
SETROPTS RACLIST(DIGTCERT,DIGTNMAP) REFRESH

/* Connect the CA certificate to the server's keyring */
RACDCERT ID(MYUSERID) CONNECT(CERTAUTH -
LABEL('LOCALCA') -
RING(ATTLS_keyring) - USAGE(certauth))
SETROPTS RACLIST(DIGTCERT,DIGTNMAP) REFRESH
```

Configuring the PowerExchange Listener on the z/OS SSL Server

In the DBMOVER configuration file, include a LISTENER statement for the port that you associated with the PowerExchange Listener task in the AT-TLS policy file. On z/OS, the LISTENER statement does not include any additional parameters for SSL processing.

You can separate PowerExchange Listeners in SSL mode and non-SSL mode. For example, you might run the PowerExchange Listener in non-SSL mode on port 13131 to connect to the PowerExchange Navigator and PowerCenter Developer, and in SSL mode on port 13132 to connect to the PowerCenter Integration Service. In this case, the AT-TLS policy file includes a rule for the Listener job using SSL on port 13132, and the DBMOVER file includes these statements:

```
LISTENER=(node1,TCPIP,13131)
LISTENER=(node1,TCPIP,13132)
```

Caution: Because PowerExchange uses Application Transparent-TLS to implement SSL on z/OS, do not include the SSL parameter in the LISTENER statement in the DBMOVER file on z/OS. In addition, do not include SSL-related statements, such as SSL_REQ_CLNT_CERT, in the file. Including SSL-related parameters or statements in the DBMOVER file causes processing to fail.

SSL Client and Server Configuration on Linux, UNIX, and Windows

To configure an SSL client or server on Linux, UNIX, or Windows, perform the following tasks:

- Optionally, verify OpenSSL operation.
- Create a CA certificate.

- Create a personal certificate.
- Customize the DBMOVER configuration file.

OpenSSL in a PowerExchange SSL Environment

OpenSSL is an open-source implementation of the SSL and TLS protocols. PowerExchange uses the OpenSSL run-time engine on Linux, UNIX, and Windows. You can also use the openssl program, which is part of the OpenSSL system, to create CA certificates and personal certificates.

On 32-bit Windows, PowerExchange includes all the components required to generate certificates: openssl.exe, ssleay32.dll, and pmlibey32.dll.

On Linux, UNIX, and 64-bit Windows, PowerExchange does not include the openssl program. However, OpenSSL is pre-installed on Linux and UNIX. To install OpenSSL on 64-bit Windows, download it from the Web site <http://www.openssl.org>

To verify that the openssl program is installed and can be used to create certificates, launch OpenSSL and issue the following command:

```
OpenSSL> version
```

OpenSSL displays the version number.

To maximize security, Informatica periodically refreshes OpenSSL to the latest version and eliminates support for weak protocols. An OpenSSL refresh might be included in a full PowerExchange release or in a hotfix.

When you create certificates using the openssl program on Linux, Unix and 64 bit Windows, the libraries used are different from the ones that PowerExchange uses to process secure connections. However, the certificates are compatible.

Creating a CA Certificate with OpenSSL

Create a CA certificate that you can use to sign personal certificates on Linux, UNIX, or Windows.

If you have a CA certificate that you can use to sign personal certificates, skip this step.

1. At the command prompt, enter the following command:

```
openssl
```

The OpenSSL> prompt appears.

2. To generate a private key and a request for a CA certificate, issue the OpenSSL req command:

```
OpenSSL> req -newkey rsa:2048 -sha1 -keyout rootkey.pem -out rootreq.pem
```

-newkey rsa:2048

Requests a new certificate request and a 2048-bit RSA private key.

-sha256

Specifies to use the SHA-256 hash function to sign the request.

-keyout rootkey.pem

File name to which to write the private key for the CA certificate.

-out rootreq.pem

File name to which to write the certificate request for the CA certificate.

3. Respond to the series of prompts that OpenSSL displays.

For most prompts, you can accept the default. For Common Name, you can use the server name.

4. To generate a public CA certificate, issue the OpenSSL x509 command:

```
OpenSSL> x509 -req -in rootreq.pem -sha1 -extensions V3_CA -signkey rootkey.pem -out  
rootcert.pem
```

-req

Specifies that the input is a certificate request, rather than a certificate.

-in rootreq.pem

Name of the input certificate request file that you created in the previous step.

-sha256

Specifies to use the SHA-256 hash function to sign the request.

-extensions V3_CA

Specifies V3_CA as the section to add certificate extensions from. Use this option to convert a certificate request into a self signed certificate using extensions for a CA.

-signkey rootkey.pem

Specifies to use the private key that you created in the previous step to sign this certificate.

-out rootcert.pem

File name to which to write the CA certificate. If you require authentication, you can specify this value for the CALIST parameter of the SSL statement in the DBMOVER file.

Creating a Personal Certificate with OpenSSL

To create a personal or subject identification certificate, issue the OpenSSL req and x509 commands, and then concatenate the two files that these commands create.

Alternatively, you can create a personal certificate using one of the following methods:

- Export an existing Windows certificate, and convert it to .pem format using OpenSSL.
- Generate a personal certificate on the mainframe using RACF, export the certificate, and convert it to .pem format using OpenSSL.

To create a personal certificate using OpenSSL, perform the following actions:

1. If the OpenSSL program is not already running, enter the following command at the command prompt:

```
openssl
```

The OpenSSL> prompt appears.

2. To generate a private key and a request for a personal certificate, issue the OpenSSL req command:

```
OpenSSL> req -newkey rsa:2048 -sha1 -keyout personalkey.pem -out subjectidreq.pem
```

-newkey rsa:2048

Requests a new certificate request and a 2048-bit RSA private key.

-sha256

Specifies to use the SHA-256 hash function (message digest) to sign the request.

-keyout personalkey.pem

File name to which to write the private key for the personal certificate.

-out rootreq.pem

File name to which to write the certificate request for the personal certificate.

3. At the prompt for a pass phrase, enter the pass phrase that you provided when you created the CA certificate.
4. At the prompt for a PEM pass phrase, enter a second pass phrase. You will specify this pass phrase in the PASS= parameter of the SSL statement in the DBMOVER configuration file.
5. Respond to the series of prompts that OpenSSL displays. You can use the same responses that you provided for the CA certificate request.
6. To generate a personal certificate, issue the OpenSSL x509 command:

```
OpenSSL> x509 -req -in personalreq.pem -sha1 -CA rootcert.pem -CAkey rootkey.pem -
CAcreateserial -out personalcert.pem
```

-req

Specifies that the input is a certificate request, rather than a certificate.

-in *personalreq.pem*

Name of the input certificate request file that you created in the previous step.

-sha256

Specifies to use the SHA-256 hash function (message digest) to sign the request.

-CA *rootcert.pem*

File name of the CA root certificate that you created in the previous step.

-CAkey *rootkey.pem*

Specifies to use the CA private key that you created in the previous step to sign this certificate.

CAcreateserial

Creates the CA serial number file.

-out *personalcert.pem*

File name to which to write the subject identification certificate.

7. Concatenate the personal key and personal certificate.

On Windows, enter the following command:

```
type personalcert.pem personalkey.pem > personalcertkey.pem
```

On Linux or UNIX, enter the following command:

```
cat personalcert.pem personalkey.pem > personalcertkey.pem
```

This step creates the file *personalcertkey.pem*. Specify this value in the KEY= parameter of the SSL statement in the DBMOVER configuration file.

Creating a Certificate By Using the z/OS RACDCERT Command

1. Choose a z/OS user ID under which to process certificates.
There is no correlation between z/OS user IDs and Linux, Unix and Windows clients.
The certificate produced will be in PKCS12DER format. This format exports both the certificate and the private key, which must be stored
2. List existing certificate information.
List existing certificate labels on the user key rings by using the TSO command RACDCERT LISTRING.
Note: You can also use the PWXUGSK utility to list existing certificates. For more information, see the *PowerExchange Utilities Guide*.

Use JCL statements such as the following ones:

```
/**
//RINGL EXEC PGM=IKJEFT01
//SYSTSPRT DD SYSOUT=*
//SYSPRINT DD SYSOUT=*
//SYSUDUMP DD SYSOUT=*
//SYSTSIN DD *
        RACDCERT LISTRING(*) ID(certificate_owner)
        END
/*
```

List the details of existing certificates for the user by using the TSO command RACDCERT LIST.

Use JCL statements such as the following ones:

```
/**
//CERTUSR EXEC PGM=IKJEFT01
//SYSTSPRT DD SYSOUT=*
//SYSPRINT DD SYSOUT=*
//SYSUDUMP DD SYSOUT=*
//SYSTSIN DD *
        RACDCERT LIST ID(certificate_owner)
        END
/*
```

If the certificate you want already exists, you can skip step 3.

Looking at existing certificates gives an idea about what to put into new certificates.

3. Run the TSO RACDCERT GENCERT command to create the certificate.

Use JCL statements such as the following ones:

```
/**
//CERTAUTH EXEC PGM=IKJEFT01
//SYSTSPRT DD SYSOUT=*
//SYSPRINT DD SYSOUT=*
//SYSUDUMP DD SYSOUT=*
//SYSTSIN DD *
        RACDCERT ID(certificate_owner) GENCERT -
        SUBJECTSDN ( -
            O('YourOrganization') -
            CN('certificate_owner.servername.yourdomain.com') -
            OU('zOS.Admin') -
            C('GB') -
        ) -
        WITHLABEL('CERTUSRPCCert') -
        SIGNWITH(CERTAUTH LABEL('LOCALCA'))
/*
```

The following table shows the mapping of the general certificate attributes to RACDCERT GENCERT parameters:

Attribute	RACDCERT GENCERT Parameter and Notes
Key size in bits	SIZE Default is 1024. Setting a value higher than 1024 might be prevented by RACF or United States export regulations.
Distinguished name	SUBJECTSDN (- O('myorganization') - CN('userid.machine.myorganization.com') - OU('myorganizationunit') C('GB') -
Message digest	default is sha1.
Expiration date	NOTAFTER(yyyy-mm-dd) Default is 12 months from the current date.
Certificate Authority	If the certificate is signed by a recognized CA, the label reflects the name of the CA. SIGNWITH (CERTAUTH LABEL (('LOCALCA'))

The following table shows z/OS-specific RACDCERT GENCERT parameters:

RACDCERT GENCERT Parameter	Usage
WITHLABEL	Used during the export of the certificate to DER format. Will be available in the PEM certificate under " Bag Attributes - friendlyName: "
SIGNWITH	Specifies the certificate with a private key that is signing the certificate. The default is to sign with the private key of the certificate being generated, thus creating a self-certified certificate. This default is appropriate for certificate authority certificates but not useful with personal certificates.

Using the z/OS TSO command RACDCERT GETCERT, the key type can be affected by the parameters PCICC, ICSF or DSA.

Use the TSO command RACDCERT LIST to check the details of the new personal certificate.

The RACDCERT GENCERT command causes the warning message to be issued:

```
IRRD175I The new profile for DIGTCERT will not be in effect until a SETROPTS
REFRESH has been issued.
```

- To refresh the RACF definitions, issue a TSO SETROPTS command such as the following one:

```
SETROPTS RACLIST(DIGTCERT,DIGTNMAP) REFRESH "
```

You need specific RACF permission to use this command.

You might omit this command if AT-TLS is not going to use the new certificate, that is, if you are generating the certificate only for use by a PowerExchange client on Linux, Unix, or Windows.

- Issue a TSO RACDCERT CONNECT command to connect the certificate to a key ring.

You might omit this command if AT-TLS is not going to use the new certificate, that is, if you are generating the certificate only for use by a PowerExchange client on Linux, Unix or Windows.

6. To export the certificate file to a PKCSK23DER format flat file, issue the TSO RACDCERT EXPORT command.

Use JCL statements such as the following ones:

```

/**
//CERTAUTH EXEC PGM=IKJEFT01
//SYSTSPRT DD SYSOUT=*
//SYSPRINT DD SYSOUT=*
//SYSUDUMP DD SYSOUT=*
//SYSTSIN DD *
        RACDCERT EXPORT(LABEL('CERTUSRPCert1')) -
                DSN('certificate_owner.PCCERT1.DER.P12') -
                PASSWORD('USRPWD') -
                FORMAT(PKCS12DER)
        END
/*

```

The following table describes the command parameters:

Parameter	Description
LABEL	Identifies the certificate. Must match the WITHLABEL parameter in the RACDCERT GENCERT command.
DSN	Identifies the output data set name in PKCS12 DER format. The data set will be dynamically allocated DCB values RECFM=VB, LRECL=84. The data set does not need to be deleted or pre-allocated.
PASSWORD	Temporary password that needs to be remembered and input to the openssl pkcs12 -clcerts command that you issue in step 8.
FORMAT	PKCS12DER

7. Perform a binary FTP of the PKCSK23DER to the Windows or Unix machine.
8. Use one of the following methods do convert the PKCSK23DER certificate to the PEM format used on the Windows or Unix machine:

- Use the PWXUSSL utility to convert the certificate to PEM format:

```

pwxussl cmd=CONVERT_CERT_PKCS12_PEM verbose=Y INFILE=E:\ MYDETAIL\SSLCerts
\Exported\certificate_owner.PCCERT1.DER.P12 pwd=pwd OUT_FILE=E:\_MYDETAIL\SSLCerts
\Exported\RACFEXPkey.pem

```

The utility issues progress and success messages:

```

PWX-37129 MONITOR statistics switched off for process PWXUSSL

Processing console program. pwxussl cmd=CONVERT_CERT_PKCS12_PEM verbose=Y
INFILE=E:\_MYDETAIL\SSLCerts\Exported\certificate_owner.PCCERT1.DER.P12 PWD=pwd
OUT_FILE=E:\_MYDETAIL\SSLCerts\Exported\RACFEXPkey.pem

Importing PKCS12 file into memory X509 objects
-----
Opening file 'E:\_MYDETAIL\SSLCerts\Exported\SSL.RACFEXP.STQA.CERT1.DER.PKCS12'
Calling d2i_PKCS12_fp()
Closing input file
Calling PKCS12_parse()
PKCS12 contains 1 CA certificates

Exporting X509 objects to PEM file

```

```

-----
Opening output file 'E:\MYDETAIL\SSLCerts\Exported\RACFEXPkey.pem'
Writing subject identification certificate
Writing Encrypted Private Key
Encoding private key using input password
Writing CA certificate 1
Closing output file

```

```

CONVERT_CERT_PKCS12_PEM ended OK

```

- Use OpenSSL to convert the certificate to PEM format:

```

c:\openSSL\bin\openssl.exe pkcs12 -clcerts -in K:\sslCertificates
\abc890_2\PCCert1\certificate_owner.PCCERT1.DER.P12 -out K:\sslCertificates
\abc890_2\PCCert1\RACFEXPkey.pem

```

```

Enter Import Password:

```

At the Enter Import Password prompt, enter the password that you specified in the TSO RACDCERT EXPORT command in step 6.

```

xxxxxxx
MAC verified OK
Enter PEM pass phrase:

```

At this prompt specify the permanent password for the certificate and private key file. This password must be specified in the PASS= parameter on the SSL statement in the DBMOVER configuration file.

```

Verifying - Enter PEM pass phrase:

```

Re-enter the permanent password.

This creates the file named in the -out parameter.

9. Enter the following parameters into the SSL statement in the DBMOVER configuration file:

SSL Parameter	Value
KEY	Output file from step 8
PASS	Permanent password from step 8

10. Test basic connectivity:

```

DTLREXE PROG=PING LOC=NODE1SSL
=>
PWX-00750 DTLREXE Input LOC=NODE1SSL, PROG=PING, PARMS=<null>, UID=<>.
PWX-00755 DTLREXE Command OK!

```

Note that with a single personal certificate in the file, it is not possible to enable verification. If verification is attempted, error messages result.

Example error messages on the PowerExchange Listener machine:

```

PWX-00591 Tasks now active = 1.
PWX-00656 Port 16495 is running in SSL mode
PWX-00652 [127.0.0.1]:1501 : TCP/IP SSL Error, rc=-1, reason <SSL_Socket_Open fa
iled: 1239336>
PWX-31023 Open secure socket failed
PWX-31045 Certificate 1 does not verify. rc=21 "X509_V_ERR_UNABLE_TO_VERIFY_LEAF
SIGNATURE".
PWX-31045 Certificate 1 does not verify. rc=27 "X509_V_ERR_CERT_UNTRUSTED".
PWX-31045 Certificate 1 does not verify. rc=20 "X509_V_ERR_UNABLE_TO_GET_ISSUER
CERT_LOCALLY".
PWX-31044 Certificate 1. Machine 'Local Client'. Type 'CA3: Self-signed X509 V1'.
Start '2014-12-11 18:37:49 GMT'. End '2042-04-27 18:37:49 GMT'. Subject '/'
emailAddress=myuid1@machine1'. Issuer '/emailAddress=myuid1@machine1'.
PWX-00591 Tasks now active = 0.

```

Example error message on the client machine:

```
DTLREXE PROG=PING LOC=pccertSSL
=>
PWX-00750 DTLREXE Input LOC=pccertSSL, PROG=PING, PARMS=<null>, UID=<>.
PWX-00752 DTLREXE Startup Error <Failed Client Connect RCs=1217/0/0>.
PWX-00652 [127.0.0.1]:1516 : TCP/IP SSL Error, rc=31045, reason <SSL_Socket_Open fa
iled: 31045>
PWX-31045 Certificate 1 does not verify. rc=21 "X509_V_ERR_UNABLE_TO_VERIFY_LEAF
_SIGNATURE".
PWX-31045 Certificate 1 does not verify. rc=27 "X509_V_ERR_CERT_UNTRUSTED".
PWX-31045 Certificate 1 does not verify. rc=20 "X509_V_ERR_UNABLE_TO_GET_ISSUER_
CERT_LOCALLY".
PWX-31044 Certificate 1. Machine z390a Type CA3: Self-signed X509 V1 Start date
100811000000Z. End date 110811235959Z. Subje
ct /C=GB/O=INFORMATICA/OU=DEVELOPMENT/
CN=certificate_owner.GBW170701.INFORMATICA.COM. Issuer
/C=GB/O=Informatica/OU=zOS.Admin/CN=irrcerta.z390a.informatica.com.
```

11. Optionally, configure for verification.

To use verification, perform the following steps:

- a. Export the z/OS CA certificate to PKCSK23DER format. You need extra RACF permission for this.
- b. Perform a binary FTP of the exported certificate.
- c. Convert the certificate to a PEM file.
- d. Create a merged file from the outputs of step 9 and the new CA PEM file.
- e. Insert the new file name into the KEY parameter of the SSL statement in the DBMOVER configuration file:

```
SSL=(KEY=
```

- f. Insert the new CA PEM file name into the CALIST parameter of the SSL statement in the DBMOVER configuration file:

```
SSL=(CALIST=
```

Customizing the DBMOVER Configuration File on the SSL Server

Customize the DBMOVER configuration file on the SSL server for SSL communication.

LISTENER Statement

The LISTENER line specifies the parameters for the PowerExchange Listener that is running in SSL mode:

```
LISTENER=(node,TCPIP,port_number,,,,,SSL)
```

Note: To avoid command failure, maintain the relative position of the SSL parameter. Six empty parameters appear between the port number and SSL parameters.

You can separate PowerExchange Listeners in SSL mode and non-SSL mode. For example, you might run the PowerExchange Listener in non-SSL mode on port 13131 to connect to the PowerExchange Navigator and PowerCenter Developer, and in SSL mode on port 13132 to connect to the PowerCenter Integration Service. In this case, the DBMOVER file includes these statements:

```
LISTENER=(node1,TCPIP,13131)
LISTENER=(node1,TCPIP,13132)
```

SSL Statement

The SSL statement specifies the SSL certificate that you use to make the SSL connection:

```
SSL=(PASS=passphrase,KEY=personalkey.pem),CALIST=calist
```

Authentication Statements

The `SSL_REQ_CLNT_CERT` statement in the `DBMOVER` file of the SSL server determines whether the server requires client authentication. When you configure an SSL server to require client authentication, the server requests the client personal certificate together with its signing CA certificates. The server checks that the personal certificate of the client is in-date and signed by a certificate authority in the CA list of the server.

Use the following syntax:

```
SSL_REQ_CLNT_CERT={N|Y}
```

When the client requires authentication of server certificates, the `SSL_ALLOW_SELFSIGNED` statement specifies whether a self-signed certificate is sufficient to authenticate the server. Use the following syntax:

```
SSL_ALLOW_SELFSIGNED={N|Y}
```

If you configure the server to require authentication of client certificates, you must make the CA certificates available to the server. Perform the following actions:

- Copy the certificates to the server machine.
- Install the certificates using the appropriate program, such as OpenSSL.
- In the `DBMOVER` file, update the `CALIST` or `CAPATH` parameter of the SSL statement to point to the CA certificates.

RELATED TOPICS:

- [“LISTENER Statement” on page 142](#)
- [“SSL Statement” on page 177](#)
- [“SSL_REQ_CLNT_CERT Statement” on page 180](#)
- [“SSL_ALLOW_SELFSIGNED Statement” on page 178](#)

Customizing the DBMOVER Configuration File on the SSL Client

Customize the `DBMOVER` configuration file on the SSL client for SSL communication.

NODE Statement

The `NODE` statement specifies the server that you want to connect to in SSL mode.

```
NODE=(server_listener,TCPIP,remote_host,port_number,,,,,{SSL|ZOSSL})
```

Use the `SSL` parameter to access a Linux, UNIX, or Windows system.

Use the `ZOSSL` parameter to access a z/OS system. However, use the `SSL` parameter, instead of the `ZOSSL` parameter, if PTFs UK26131 (z/OS 1.8) or UK26132 (z/OS 1.9) have been installed on the z/OS machine. These PTFs rectify APAR PK46403.

Note: To avoid command failure, maintain the relative position of the `SSL` or `ZOSSL` parameter. Five empty parameters appear between the port number parameter and the `SSL` or `ZOSSL` parameter.

SSL Statement

The `SSL` statement specifies the SSL key, pass phrase, and Certificate Authority list (`CALIST`) that you are using to make the SSL connection. For example:

```
SSL=(PASS=passphrase,KEY=personalkey.pem,CALIST=root.pem)
```

You can replace `CALIST` with `CAPATH`. For example:

```
SSL=(PASS=client,KEY=client.pem,CAPATH=/pwx/certs/)
```

To optimize performance, specify the location of multiple certificates with `CAPATH`.

Use `CAPATH` to specify the trusted CA directory of the OpenSSL installation on Linux or UNIX.

Use CAPATH to specify the 'certs' directory of the OpenSSL installation on Windows.

Authentication Statements

The SSL_REQ_SRVR_CERT statement in the DBMOVER file of the SSL client determines whether the client performs server authentication. When you configure an SSL client to perform server authentication, the client checks that the personal certificate of the server is in-date and signed by a Certificate Authority in the CA list of the client.

Use the following syntax:

```
SSL_REQ_SRVR_CERT={N|Y}
```

When the client authenticates server certificates, the SSL_ALLOW_SELFSIGNED statement specifies whether a self-signed certificate is sufficient to authenticate the server. Use the following syntax:

```
SSL_ALLOW_SELFSIGNED={N|Y}
```

If you configure the client to perform authentication of server certificates, you must make the CA certificates available to the client. Perform the following actions:

- Copy the certificates to the client machine.
- Install the certificates using the appropriate steps for the operating system. Refer to the documentation for the client operating system for information about installing certificates.
- In the DBMOVER file, update the CALIST or CAPATH parameter of the SSL statement to point to the CA certificates on the client.

RELATED TOPICS:

- [“NODE Statement” on page 155](#)
- [“SSL Statement” on page 177](#)
- [“SSL_REQ_SRVR_CERT Statement” on page 181](#)
- [“SSL_ALLOW_SELFSIGNED Statement” on page 178](#)

Verifying Connections

After you configure PowerExchange, you can ping the remote PowerExchange Listener to verify the connection.

From a PowerExchange SSL client machine, issue the following DTLREXE command:

```
DTLREXE PROG=PING LOC=sslnode
```

If the connection is working, the Listener issues the following message:

```
PWX-00755 DTLREXE Command OK!
```

CHAPTER 8

PowerExchange Alternative Network Security

This chapter includes the following topics:

- [PowerExchange Alternative Network Security Overview, 320](#)
- [Modes of PowerExchange Encryption, 320](#)
- [Setting Encryption Defaults in the DBMOVER Configuration File, 321](#)

PowerExchange Alternative Network Security Overview

PowerExchange provides alternatives to Secure Socket Layer (SSL) network security. In addition to an internal proprietary encryption method, PowerExchange supports AES encryption.

To enable alternative network security, define the ENCRYPT statement in the DBMOVER configuration file on the system where the data movement process runs. If you define ENCRYPT=AES, also define the ENCRYPTLEVEL statement. Also, you can define the **Encrypt Level** and **Encrypt Type** connection attributes to override the values defined in the DBMOVER statements for individual connections.

Note: Informatica recommends that you use Secure Sockets Layer (SSL) authentication instead of configuring the **Encrypt Level** and **Encryption Type** connection attributes or the ENCRYPT and ENCRYPTLEVEL statements in the DBMOVER configuration file. SSL authentication provides stricter security and is used by other Informatica products.

Modes of PowerExchange Encryption

PowerExchange alternative network security supports the following modes of data encryption when moving data:

- No encryption
- PowerExchange encryption
- AES encryption

PowerExchange Encryption Standard

This is a PowerExchange proprietary encryption algorithm.

AES Encryption Standard

The Advanced Encryption Standard (AES) is a specification for the encryption of electronic data established by the U.S. National Institute of Standards and Technology (NIST) in 2001.

AES is a subset of the Rijndael family of ciphers. AES includes three members of the Rijndael family, each with a block size of 128 bits but different key lengths of 128, 192 and 256 bits.

AES has been adopted by the United States government and is now used worldwide. It supersedes the Data Encryption Standard (DES). The algorithm described by AES is a symmetric-key algorithm. That is, the same key is used for both encrypting and decrypting the data.

The federal standard document FIPS PUB 197 (FIPS 197) describes this algorithm.

Setting Encryption Defaults in the DBMOVER Configuration File

To set the encryption that PowerExchange uses by default, define ENCRYPT and ENCRYPTLEVEL statements in the DBMOVER configuration file on the system where the data movement process runs. Also, to ensure predictable results, define these statements with the same values on the client machine.

Note: To enable encryption for a PowerCenter session or a Data Quality mapping, set the **Encryption Type** and **Encryption Level** connection attributes. These connection attributes override the ENCRYPT and ENCRYPTLEVEL statements in the DBMOVER configuration file on the PowerCenter Integration Service or Data Integration Service machine.

CHAPTER 9

PowerExchange Nonrelational SQL

This chapter includes the following topics:

- [PowerExchange Nonrelational SQL Overview, 322](#)
- [PowerExchange Nonrelational SQL Syntax, 323](#)
- [DTLDESCRIBE Metadata Syntax, 326](#)
- [DTLDESCRIBE Qualifiers, 342](#)
- [Restrictions on PowerExchange Nonrelational SQL, 344](#)
- [Truncating a Target Table, 345](#)
- [Inserting Rows into a Table, 345](#)

PowerExchange Nonrelational SQL Overview

To access nonrelational sources, such as an IMS database or VSAM file, use the PowerExchange Navigator to create a data map that defines various physical and logical records. PowerExchange accepts a SQL statement in the normal way from the client and passes it to the PowerExchange Listener. The PowerExchange Listener accesses the database and gets the rows. PowerExchange accepts only a subset of the extensive SQL syntax. You must use only the SQL that PowerExchange supports for the SQL requests to be accepted.

The places where you can enter nonrelational SQL include:

- **SQL Statement** box in the **Database Row Test** dialog box
- **Pre SQL** and **Post SQL** session properties for a nonrelational source or target
- **Filter Override** session property for the source
- Lookup transformations
- PowerCenter source qualifier

PowerExchange Nonrelational SQL Syntax

PowerExchange nonrelational data sources support SELECT, INSERT, UPDATE, DELETE, and TRUNCATE statements.

The basic syntax of a SELECT statement that is supported by nonrelational data sources is as follows:

```
select col_a, col_b, ... col_n from schema.map_name.table_name [where condition]
```

The *schema.map_name* is the data map name based on the data-map naming convention. The *table_name* is a user-defined name that is defined in the data map.

The WHERE clause is optional. You can construct the WHERE *condition* from one or more clauses that specify simple comparisons or other types of conditions. Join these clauses by using one or more of the Boolean operators AND, OR, and NOT. If you do not use parentheses to indicate the order of evaluation, the default order of NOT > AND > OR is used.

A simple comparison is one of the following types:

- basic
- like
- between
- null

SQL Syntax for a Basic Comparison

In a basic comparison, a column is compared with either another column or a constant.

For example:

```
col1 = '123'
```

Complex expressions such as (col1 + col2) or (col1 * 3) are *not* supported.

The following table lists the valid operators:

Name	Operator
=	Equals
<>	Not equal
!=	Not equal
>	Greater than
>=	Not less than
<	Less than
<=	Not greater than

- Character constants must be enclosed in single quotation marks and not exceed 255 characters in length.
- Binary constants must be enclosed in single quotation marks and preceded by "X," for example, X'00'.
- Numeric constants can be specified with an optional leading sign and decimal point. Exponential floating point format is allowed.

Comparisons can be made only between operands of similar types:

- A numeric operand can be compared only with another numeric operand.
- A character operand can be compared with either a character or binary operand.
- A binary operand can be compared with either a character or binary operand.
- Date, time and timestamp (date-time) operands can be compared only with an identical column type or a character constant in the correct format. For dates and timestamps, the year value must be in the valid range of years that is set by the DATERANGE statement in the DBMOVER configuration file.

Date-time constants must have the following formats:

- DATE

YYYY-MM-DD

Single-digit *DD* and *MM* values are allowed. If the lower order portion of the date is omitted, it defaults to the values specified as the **Data Defaults** within the data map.

- TIME

HH.MI.SS

Single-digit *HH*, *MI*, and *SS* values are also allowed. If the lower order portion of the time is omitted it defaults to the following values: *MI*=00, *SS*=00.

- TIMESTAMP

YYYY-MM-DD HH.MI.SS.NNNNNN

The *NNNNNN* value is the fractional seconds value. Single digits are allowed for *DD*, *MM*, *HH*, *MI*, *SS*, and *NNNNNN*. The date and time values default to the following values: *MI*=00, *SS*=00, *HH*=00, *NNNNNN*=000000.

- You can enter only the date or only the time component with the other component defaulting. If insufficient information is available to deduce which component has been entered, it will be assumed to be the date component.

When character fields of different length are compared, the operands are made equal lengths by padding the shorter one with spaces, prior to the comparison being made.

Similarly for binary fields, when fields of different length are compared, the operands are made equal lengths by padding the shorter one with nulls (X'00'), prior to the comparison being made.

Comparisons involving character or binary columns are limited to columns less than 256 bytes in length.

LIKE Comparison

This comparison is a pattern-matching comparison. The comparison is true if the column name matches the pattern.

Example:

```
coll [NOT] LIKE pattern
```

The pattern is a character constant of 255 characters or less that contains any combination of characters within which the percent sign (%) and the underscore character (_) have special significance:

- % represents a series of zero or more arbitrary characters. Entering %% is the same as entering %.
- _ represents exactly one arbitrary character.
- Any character other than % or _ represents a single occurrence of that character.
- Trailing blanks in the column or in the pattern will be ignored for comparison purposes.

The following table lists example patterns, column names that match the pattern, and column names that do not match the pattern:

Pattern	Matches	Does Not Match
%TAB%	TAB, ATAB, TABLE, TABULAR, MY_TABLE	MY_TAABLE
TA%	TAB, TABLE, TABULAR	ATAB, MY_TABLE, MY_TAABLE
TA_LE	TABLE	TAB , ATAB, TABULAR, MY_TABLE, MY_TAABLE

Use an ESCAPE clause to define the escape character to be used to escape the “%” and “_” characters in the LIKE statement so that they can be matched explicitly.

You can use ESCAPE '\ ' in order that a string with 'A%_ ' would allow all values starting with 'A' and ending with '_ ' to be returned.

For example, the pattern %_TAB% matches the column name MY_TABLE but does not match the column name MYTABLE.

BETWEEN Comparison

This comparison is testing a column for being falling in side or outside (used with NOT) a range of values.

Example:

```
col1 [NOT] BETWEEN valueA AND valueB
```

In this example valueA and valueB could be another column or a constant of the same type as the column being compared against. It is equivalent to coding two simple conditions as follows:

```
[NOT] (col1 >= valueA AND col1 <= valueB)
```

NULL Comparison

This comparison is testing a column for being NULL or NOT NULL. It is treated as though it is less than any other possible value therefore it cannot be equal to anything other than another NULL column.

Example:

```
col1 IS [NOT] NULL
```

Allowed Keywords in the SQL Syntax

Examples of allowable keywords are:

```
ALL  
DISTINCT
```

Accessing Data with Column Names the Same as SQL Keywords

You can access data with column names the same as SQL keywords by using single quotes around the column name. For example:

```
SELECT 'DISTINCT' FROM TABLE
```

Using SQL with User-Defined Fields

You should not use the SQL keyword 'DISTINCT' if the data map includes user-defined fields. Attempting to use DISTINCT with user defined fields causes a message in the logfile 'SELECT DISTINCT not guaranteed with Expressions'.

If DISTINCT is essential to a query then the best approach is to create another table in the data map that contains only the fields that are actually required for that query.

IMS Call Considerations

The potential to store data in a non-sequential order in HDAM, DEDB, and PHDAM databases causes complications when dealing with ranges of data. Ideally, if searching for values between valueA and valueB, the search starts at valueA and issues Get Next calls until valueB. This sort of processing is possible only in an indexed sequential format. Where the keys are not in sequential order, positioning on valueA might be beyond some candidate segments. In this case, the result set will be incomplete. The following example shows the problem:

```
Select * from IMSSC1.IMSMPL.IMSDEMOS Where KEY >= 'A' and KEY <= 'D'
```

In this example, the randomizer has placed the records physically in the following key sequence:

```
C  A  D  E  B  H  F
```

Positioning on the first value, in this case A, then issuing Get Next calls until the end value of D would return only A and D, with candidate value C physically residing in the data set before A, and B after D.

The same issue arises for SELECT statements such as:

```
Select * from IMSSC1.IMSMPL.IMSDEMOS Where KEY = 'D' or KEY = 'B' or KEY = 'A'
```

In this case, three GU requests are needed instead of one GN request with a range (\geq 'B' and \leq 'D'). In case of the range request, IMS positions on the 'B' segment and reads by means of the twin pointer forward and misses the 'C' and 'D' segment because they are physically stored before the 'B' segment. The number of GU requests depends on the number of predicates in the query.

Therefore, if a range is requested of a HDAM, DEDB or PHDAM database, PowerExchange is forced to process the complete database sequentially, selecting all the required records as they are found.

DTLDESCRIBE Metadata Syntax

PowerExchange provides a special SQL syntax, called DTLDESCRIBE, for accessing metadata. The DTLDESCRIBE syntax provides qualifiers. The meaning of these qualifiers depends on the source database.

DTLDESCRIBE TABLES Qualifier

Enables you to list the available tables.

Use the following syntax:

```
dtldescribe tables, [qualifier1], [qualifier2], [table_name], [comments], [ORDER],  
[escape_character], [respect_case1], [respect_case2], [respect_case3],  
[extended_information], [access_method]
```

After the basic syntax of dtldescribe tables, all of the parameters in brackets [] are optional. If any of the parameters are coded, you must include the intervening commas so that the parser can determine which field is included.

The following table describes the parameters:

Parameter	Description
<i>qualifier1, qualifier2</i>	Returns the first and second data map qualifiers.
<i>table_name</i>	The primary table qualifier. Returns the table name. If you leave this parameter blank, all tables are returned.
<i>comments</i>	Set this parameter to Y to return any comments found in data in the database.
ORDER	Leave this field blank except for a comma, which is required to maintain position of subsequent fields.
<i>escape_character</i>	Specifies the character to be used as the escape character. Default is the ~ character.
<i>respect_case1, respect_case2, respect_case3</i>	Set these parameters to Y to respect the case of the <i>qualifier1</i> , <i>qualifier2</i> and <i>table_name</i> values, respectively. Or leave these parameters blank.
<i>extended_information</i>	Set this parameter to Y to return the following fields that contain extra information. Applicable to NRDB, NRDB2, and CAPX only. Default is N, which indicates that no extended information is required.
<i>access_method</i>	<p>For NRDB, NRDB2, CAPX, and CAPXRT only:</p> <p>Specifies the valid access method or methods for which information is returned. The results of the DTLDESCRIBE are filtered to return information for only the specified access methods. Use this parameter to return a list of data maps that are applicable for certain access methods. Any number or combination of access method identifiers can be used, such as, OS or AKT. If no value is specified, the DTLDESCRIBE returns information for all access methods by default.</p> <ul style="list-style-type: none"> - ADABAS - A - CAPX / RT - C - DATACOM - X - DB2 - Z - DB2390IMG - G - DB2UNLD - W - DL1 - D - ESDS - E - IDMS - I - KSDS - K - MQ - Q - ODBA - O - RRDS - N - SEQ - S - TAPE - T - USER - U <p>The following extraction map access methods are applicable only to CAPX and CAPXRT:</p> <ul style="list-style-type: none"> - XDB2 - B - XMSSQL - L - XORACLE - P

RELATED TOPICS:

- [“DTLDESCRIBE Qualifiers” on page 342](#)

DTLDESCRIBE TABLES Returned Information

The following table describes the columns:

Column Name	Description	Extended information
qualifier_1	Creator/Schema name.	No
qualifier_2	PowerExchange Data Map name will be returned, or null for NRDB2.	No
tablename	Return table name.	No
comments	Comments column always present. Comment information is included only if requested by specifying the comments indicator in the describe.	No
type	Type of object for which metadata was returned, such as TABLE or MATERIALIZED VIEW.	No
Acc_mth	The access method (from data map)	Yes
Acc_mths01	Adabas	Database ID
Acc_mths01	SEQ, KSDS, ESDS	MapFileName
Acc_mths01	IDMS	MapSubSchemaName
Acc_mths01	ODBA or DL/I	DBDName
Acc_mths01	Datacom	Database ID
Acc_mths01	CAPX or CAPXRT	The access method of the original data map
Acc_mths01	Other access methods	NULL
Acc_mths02	Adabas	File Number
Acc_mths02	ODBA or DL/I	DBDType
Acc_mths02	Datacom	Table record length
Acc_mths02	IDMS or VSAM	C (if table is valid for data capture) or NULL
Acc_mths02	CAPX	Condense option (Full, Part or None)
Acc_mths02	DB2UNLD	Null indicator value
Acc_mths02	Other access methods	NULL
Acc_mths03	Datacom	Table ID

Column Name	Description	Extended information
Acc_mths03	IDMS or VSAM	Database name
Acc_mths03	CAPX or CAPXRT	Creator of the capture registration
Acc_mths03	DB2	DB2 Sub system ID
Acc_mths03	DB2UNLD	Unload type
Acc_mths03	Other access methods	Not Used. (comma needed to maintain position of later fields)
Acc_mths04	Datacom	Version
Acc_mths04	CAPX or CAPXRT	Name of the registration's source table
Acc_mths04	DB2 or DB2UNLD	DB2 table name
Acc_mths04	Other access methods	Not Used. (comma needed to maintain position of later fields)
Acc_mths05	Datacom	Recovery (Y/N)
Acc_mths05	DB2UNLD	Unload file name
Acc_mths05	Other access methods	Not Used. (comma needed to maintain position of later fields)
Acc_mths05	Base Record	Describes the dependencies for a table. For example, ROOT (SEG2 (SEG3, SEG4) , SEG5) For Datacom, Base Record returns the short table name.

RELATED TOPICS:

- [“DTLDESCRIBE Qualifiers” on page 342](#)

DTLDESCRIBE PROCEDURES Qualifier

Enables you to list the available stored procedures:

```
dtldescribe procedures, [qualifier1], [qualifier2], [procedure_name],
[escape_character], [respect_case1], [respect_case2], [respect_case_procname]
```

After the basic syntax of `dtldescribe procedures`, all of the parameters in brackets [] are optional. If any of the parameters are coded, you must include the intervening commas so that the parser can determine which field is included.

The following table describes the parameters:

Parameter	Description
<i>qualifier1</i>	Catalog name.
<i>qualifier2</i>	Schema name.

Parameter	Description
<i>procedure_name</i>	Procedure name
<i>escape_character</i>	Specifies the character to be used as the escape character. Default is ~.
<i>respect_case1</i> , <i>respect_case2</i>	Set these parameters to Y to respect the case of the <i>qualifier1</i> and <i>qualifier2</i> parameters, respectively. Or leave these parameters blank.
<i>respect_case_procname</i>	Set this parameter to Y to respect the case of the <i>procedure_name</i> value. Or leave this parameter blank.

DTLDESCRIBE PROCEDURES Returned Information

The following table describes the columns:

Column Name	Description
PROCEDURE_CAT	Procedure Catalog, null if not present; VARCHAR.
PROCEDURE_SCHEM	Procedure Schema, null if not present; VARCHAR.
PROCEDURE_NAME	Procedure Name, not null; VARCHAR.
NUM_INPUT_PARAMS	Number of input parameters, not null (N/A for SQL Server); INTEGER.
NUM_OUTPUT_PARAMS	Number of output parameters, not null (N/A for SQL Server); INTEGER.
NUM_RESULT_SETS	Number of result sets, not null (N/A for SQL Server); INTEGER.
REMARKS	Comments; VARCHAR.
PROCEDURE_TYPE	Procedure type.

DTLDESCRIBE PROCEDURES is implemented for DB2400C and MSSQL.

DTLDESCRIBE PROCEDURECOLUMNS Qualifier

Enables you to list the available columns and information for given stored procedures:

```
dtldescribe procedurecolumns, [qualifier1], [qualifier2], [procedure_name],
[column_name], [escape_character], [respect_case1], [respect_case2],
[respect_case_procname], [respect_case_colname]
```

After the basic syntax of `dtldescribe procedurecolumns`, all of the parameters in brackets `[]` are optional. If any of the parameters are coded, you must include the intervening commas so that the parser can determine which field is included.

The following table describes the parameters:

Parameter	Description
<i>qualifier1</i>	Catalog name.
<i>qualifier2</i>	Schema name.
<i>procedure_name</i>	Procedure name.
<i>column_name</i>	Column name.
<i>escape_character</i>	Specifies the character to be used as the escape character. Default is the ~ character.
<i>respect_case1, respect_case2</i>	Set these parameters to Y to respect the case of the <i>qualifier1</i> and <i>qualifier2</i> values, respectively. Or leave these parameters blank.
<i>respect_case_procname</i>	Set this parameter to Y to respect the case of the <i>procedure_name</i> value. Or leave this parameter blank.
<i>respect_case_colname</i>	Set this parameter to Y to respect the case of the <i>column_name</i> value. Or leave this parameter blank.

DTLDESCRIBE PROCEDURECOLUMNS Returned Information

The following table describes the columns:

Column Name	Description
PROCEDURE_CAT	Procedure catalog, null if not present; VARCHAR
PROCEDURE_SCHEM	Procedure schema, null if not present; VARCHAR
PROCEDURE_NAME	Procedure name; VARCHAR
COLUMN_NAME	Column name; VARCHAR
COLUMN_TYPE	Column type, not null; SMALLINT Can be 0 - SQL_PARAM_TYPE_UNKNOWN 1 - SQL_PARAM_INPUT - input parameter 2 - SQL_PARAM_INPUT_OUTPUT - input/output parameter 3 - SQL_RESULT_COLUMN - Parm is a column inset 4 - SQL_PARAM_OUTPUT - output parameter 5 - SQL_RETURN_VALUE - Column is return value of procedure
DATA_TYPE	SQL datatype, not null; SMALLINT
TYPE_NAME	Type name, character string representing datatype, not null; VARCHAR
COLUMN_SIZE	Column size; INTEGER
BUFFER_LENGTH	Buffer length. Maximum number of bytes required to store column data; INTEGER

Column Name	Description
DECIMAL_DIGITS	Scale, NULL if not applicable; SMALLINT
NUM_PREC_RADIX	Precision; SMALLINT
NULLABLE	Nullable, not null, determines if column will accept a null value; SMALLINT. Can be 0 - SQL_NO_NULLS 1 - SQL_NULLABLE
REMARKS	Remarks; VARCHAR
COLUMN_DEF	Column default value; VARCHAR
SQL_DATA_TYPE	SQL Datatype, not null; SMALLINT
SQL_DATETIME_SUB	Datetime subtype. Can be 1 - SQL_CODE_DATE 2 - SQL_CODE_TIME 3 - SQL_CODE_TIMESTAMP
CHAR_OCTET_LENGTH	Maximum length in bytes of a character type column; INTEGER
ORDINAL_POSITION	Ordinal Position if parameter given by Column name in result set, not null; INTEGER
IS_NULLABLE	Can the column contain nulls. "YES" if column can contain nulls, "NO" if not; VARCHAR

DTLDESCRIBE PROCEDURECOLUMNS is implemented for DB2400C and MSSQL.

DTLDESCRIBE COLUMNS Qualifier

Enables you to list the available columns:

```
dtldescribe columns, [qualifier1], [qualifier2], [table_name], [comments], [ORDER],
[escape_character], [respect_case1], [respect_case2], [respect_case3],
[extended_information], [access_method]
```

After the basic syntax of `dtldescribe columns`, all of the parameters in brackets [] are optional. If any of the parameters are coded, you must include the intervening commas so that the parser can determine which field is included.

The following table describes the parameters:

Parameter	Description
<i>qualifier1</i>	-
<i>qualifier2</i>	Always NULL if the <code>dtldescribe</code> operation is done by using NRDB2. Otherwise, this parameter returns the PowerExchange data map name.
<i>table_name</i>	The primary table qualifier. Returns the table name. If this parameter is blank, returns columns for all available tables.

Parameter	Description
<i>comments</i>	If set to Y this will return any comments data found in the database.
ORDER	Optional argument, Y will force order by column number or blank will leave column order undefined.
<i>escape_character</i>	Specifies the character to be used as the escape character. Default is ~.
<i>respect_case1</i> <i>respect_case2</i> <i>respect_case3</i>	Set to Y this will respect the case of qualifier1, qualifier2 and tablename respectively - alternative is blank.
<i>extended_information</i>	Set this parameter to Y to return additional information that depends on the data access method. Applicable to NRDB, NRDB2, DB2 for z/OS, and DB2 for i5/OS only. Defaults to N, which indicates no extended information is required.
<i>access_method</i>	For NRDB, NRDB2, CAPX, and CAPXRT only. Specifies the valid access method or methods for which information is returned. The results of the DTLDESCRIBE are filtered to return information only for the specified access methods. Use this method to return a list of data maps applicable for certain access methods. You can use any number or combination of access method identifiers. If you specify no value, the DTLDESCRIBE returns all access methods. <ul style="list-style-type: none"> - ADABAS - A - CAPX / RT - C - DATACOM - X - DB2 - Z - DB2390IMG - G - DB2UNLD - W - DL1 - D - ESDS - E - IDMS - I - KSDS - K - MQ - Q - ODBA - O - RRDS - N - SEQ - S - TAPE - T - USER - U

RELATED TOPICS:

- [“DTLDESCRIBE Qualifiers” on page 342](#)

DTLDESCRIBE COLUMNS Returned Information

The following table describes the columns:

Column Name	Description	Extended Information
Qualifier1	Creator/Schema name.	No
Qualifier2	PowerExchange data map name will be returned, or null for NRDB2.	No
Tablename	Return table name.	No
Column_name	Column name.	No
Type	Field format.	No
Precision	Field length.	No
Scale	Decimal places.	No
Nullable	Nullable.	No
Detail_type	Internal column type representation.	No
Comments	Comments column always present. Comment information is included only if requested by specifying the comments indicator in the describe.	No
Key_type	Reserved for future use.	No
Ref_qualifier_1	Reserved for future use.	No
Ref_qualifier_2	Reserved for future use	No
Ref_table_name	Reserved for future use	No
Ref_column_name	Reserved for future use	No
Base Record	Describes the dependencies for a table. For example, <code>ROOT (SEG2 (SEG3, SEG4) , SEG5) .</code>	NRDB
Base Field	Column base field, such as <code>ROOT:ROOTKEY.</code>	NRDB
Base Field Type	Field type of the column's base field (CHAR, VARCHAR etc.).	NRDB
Base Field Offset	Returns the offsets of the field within the record. There is a case when no offset is returned. If you have an array of fields and you collapse the array when you generate the table the offset of those fields will be 0.	NRDB
Field Usage	Usage of field. For example, COMP or DISPLAY for Cobol.	NRDB
Field Level	Indent level of the field in the Cobol copybook.	NRDB

Column Name	Description	Extended Information
Original Field Name	Original name of the field as given in the Cobol copybook.	NRDB
Field Picture	Cobol picture format of the field.	NRDB
CCSID	CCSID.	DB2
Internal_cp	Internal code page.	DB2, NRDB, NRDB2, CAPX, and CAPXRT
Codepage_name	Code page name.	DB2, NRDB, NRDB2, CAPX, and CAPXRT
Powercenter_name	PowerCenter name of the code page.	NRDB, NRDB2, CAPX, and CAPXRT

RELATED TOPICS:

- [“DTLDESCRIBE Qualifiers” on page 342](#)

DTLDESCRIBE RECORDS Qualifier

Enables you to list the available columns. Applicable to NRDB and NRDB2.

```
dtldescribe records, [qualifier1], [qualifier2], [table_name], [comments], [ORDER],
[escape_character], [respect_case1], [respect_case2], [respect_case3],
[extended_information], [access_method]
```

After the basic syntax of `dtldescribe records`, all of the parameters in brackets [] are optional. If any of the parameters are coded, you must include the intervening commas so that the parser can determine which field is included.

The following table describes the parameters:

Parameter	Description
<i>qualifier1, qualifier2</i>	Creator or schema name.
<i>table_name</i>	The primary table qualifier. If this parameter is blank, returns records for all available tables.
<i>comments</i>	Set this parameter to Y to return any comments found in data in the database.
ORDER	This field must be left blank except for a comma, which is needed to maintain the position of subsequent fields.
<i>escape_character</i>	Specifies the character to be used as the escape character. Default is the ~ character.
<i>respect_case1, respect_case2, respect_case3</i>	Set these parameters to Y to respect the case of the <i>qualifier1</i> , <i>qualifier2</i> and <i>table_name</i> values, respectively. Or leave these parameters blank.

Parameter	Description
<i>extended_information</i>	Set this parameter to Y to returns additional information that depends on the data access method. Applicable to NRDB, NRDB2, DB2 for z/OS, and DB2 for i5/OS only. Defaults to N, which indicates that no extended information is required.
<i>access_method</i>	For NRDB,NRDB2, CAPX, and CAPXRT only: Specifies the valid access methods for which information is returned. The results of the DTLDESCRIBE are filtered to return information for only the specified access methods. This is useful to return a list of data maps that are applicable for certain access methods. Any number or combination of access method identifiers can be used, such as OS or AKTM. If no value is used, the DTLDESCRIBE defaults to return all access method. <ul style="list-style-type: none"> - ADABAS - A - CAPX / RT - C - DATACOM - X - DB2 - Z - DB2390IMG - G - DB2UNLD - W - DL1 - D - ESDS - E - IDMS - I - KSDS - K - MQ - Q - ODBA - O - RRDS - N - SEQ - S - TAPE - T - USER - U

DTLDESCRIBE RECORDS Returned Information

DTLDESCRIBE RECORDS returns fields containing the following information. All fields defaults to N which means that no extended information required.

The following table describes the columns:

Field	Returned information
Record name	Name of the record.
Acc_mths01	IDMS - IDMS Record Name. ODBA or DL/1 - RecSegName. Other access methods - NULL.
Acc_mths02	ADABAS - File Number. ODBA or DL/1 - DBDType. Other access methods - NULL.

Field	Returned information
Acc_mths03	Not used. Reserved for future use. The comma is needed to maintain position of subsequent fields.
Acc_mths04	Not used. Reserved for future use. The comma is needed to maintain position of subsequent fields.
Acc_mths05	IDMS Log-Based. The technique that IDMS uses to physically store occurrences of the record type. <ul style="list-style-type: none"> - C = CALC - D = DIRECT - V = VIA - VS = VSAM - VSC - VSAM CALC
Acc_mths06	IDMS - Record ID.
Acc_mths07	IDMS - Compressed.
Acc_mths08	IDMS - Variable.
Acc_mths09	IDMS - Page Group.
Acc_mths10	IDMS - Radix.
Acc_mths11	IDMS - Area Name.
Acc_mths12	IDMS - Minimum Root length.
Acc_mths13	IDMS - Data length.
Acc_mths14	IDMS - Prefix length.
Acc_mths15	IDMS - Control Portion length.
Source Filename	Contains the file name of the Copybook. Not currently implemented.
DB Filename	Contains the file name of the DBD source file. Not currently implemented.

DTLDESCRIBE PKEYS Qualifier

```
dtldescribe pkeys, [qualifier1], [qualifier2], [table_name], [comments], [ORDER],
[escape_character], [respect_case1], [respect_case2], [respect_case3],
[extended_information], [access_method]
```

After the basic syntax of `dtldescribe pkeys`, all of the parameters in brackets `[]` are optional. If any of the parameters are coded, you must include the intervening commas so that the parser can determine which field is included.

The following table describes the parameters:

Parameter	Description
<i>qualifier1, qualifier2</i>	Returns the first and second data map qualifiers.
<i>table_name</i>	The primary table qualifier. Returns the table name.
<i>comments</i>	Set this parameter to Y to return any comments found in data in the database.
ORDER	This field must be left blank except for a comma, which is needed to maintain position of subsequent fields.
<i>escape_character</i>	Specifies the character to be used as the escape character. Default is the ~ character.
<i>respect_case1, respect_case2, respect_case3</i>	Set these parameters to Y to respect the case of the <i>qualifier1, qualifier2</i> and <i>table_name</i> parameters, respectively. Or leave these parameters blank.
<i>extended_information</i>	This field must be left blank except for a comma, which is needed to maintain position of subsequent fields.
<i>access_method</i>	<p>For NRDB, NRDB2, CAPX, and CAPXRT only:</p> <p>Specifies the valid access method or methods for which information is returned. The results of the DTLDESCRIBE are filtered to return information for only the specified access methods. Use this parameter to return a list of data maps that are applicable for certain access methods. Any number or combination of access method identifiers can be used, such as OS or AKT. If no value is specified, the DTLDESCRIBE returns information for all access methods by default.</p> <ul style="list-style-type: none"> - ADABAS - A - CAPX / RT - C - DATACOM - X - DB2 - Z - DB2390IMG - G - DB2UNLD - W - DL1 - D - ESDS - E - IDMS - I - KSDS - K - MQ - Q - ODBA - O - RRDS - N - SEQ - S - TAPE - T - USER - U

PKEYS is implemented for:

- ADABAS
- DB2
- DB2390IMG
- DB2400C

- DB2UDB
- MSSQL
- NRDB / NRDB2 (only for DL/I, ODBA, KSDS and Datacom data maps)
- ORACLE

RELATED TOPICS:

- [“DTLDESCRIBE Qualifiers” on page 342](#)

DTLDESCRIBE FKEYS Qualifier

```
dtldescribe fkeys, [pk_qualifier1], [pk_qualifier2], [pk_tablename], [fk_qualifier1],
[fk_qualifier2], [fk_tablename], [comments], [ORDER], [escape_character],
[respect_case_pk_1], [respect_case_pk_2], [respect_case_pk_3], [respect_case_fk_1],
[respect_case_fk_2], [respect_case_fk_3], [extended_information], [access_method]
```

After the basic syntax of `dtldescribe fkeys`, all of the parameters in brackets [] are optional. If any of the parameters are coded, you must include the intervening commas so that the parser can determine which field is included.

The following table describes the parameters:

Parameter	Description
<i>pk_qualifier1, pk_qualifier2</i>	Primary key qualifiers.
<i>pk_tablename</i>	This is the primary table qualifier.
<i>fk_qualifier1, fk_qualifier2</i>	These refer to foreign key qualifiers.
<i>fk_tablename</i>	The foreign table qualifier.
<i>comments</i>	Set this parameter to Y to get any comments found in data in the database.
ORDER	This field must be left blank except for a comma, which is needed to maintain position of subsequent fields.
<i>escape_character</i>	Specifies the character to be used as the escape character. Default is ~.
<i>respect_case pk_1, respect_case pk_2, respect_case pk_3, respect_case fk_1, respect_case fk_2, respect_case fk_3</i>	Set these parameters to Y to respect the case of the <i>pk_qualifier1</i> , <i>pk_qualifier2</i> , <i>pk_tablename</i> , <i>fk_qualifier1</i> , <i>fk_qualifier2</i> , and <i>fk_tablename</i> values, respectively. Or leave these parameters blank.
<i>extended_information</i>	This field must be left blank except for a comma, which is needed to maintain position of subsequent fields.
<i>access_method</i>	This field must be left blank except for a comma, which is needed to maintain position of subsequent fields.

FKEYS is implemented for:

- DB2
- DB2UDB
- DB2400C

- ORACLE
- DB2390IMG
- MSSQL

RELATED TOPICS:

- [“DTLDESCRIBE Qualifiers” on page 342](#)

DTLDESCRIBE FKEYS General Notes

For metadata requests, the qualifiers will, as a default, be converted to the case that is standard on the source database, for example, uppercase for Oracle. Hence, supplying a qualifier1 field of scott or SCOTT will produce the same results because both will be treated as uppercase.

Alternatively, by setting the Respect case 1 option to Y, this defaulting to the database case will not be done. Hence a qualifier1 of scott will produce no results, whereas SCOTT will be successful.

Respect case 2 is relevant for qualifier2 and Respect case 3 is relevant for tablename.

The metadata qualifiers support wildcards as follows:

- * - one or more matching characters
- ? - a single matching character

If either of the wildcard characters are used within a column or table name, precede each occurrence of them with the escape character.

For example, a request for tab* would list all tables beginning with tab, whereas a request for tab~* would list only the table that was named tab*.

Therefore, to list only tables called s*ott in lowercase and return comments this SQL could be used:

```
dtldescribe tables,s~*ott,,,Y,,,Y
```

DTLDESCRIBE FKEYS Table Information Retrieval

For example, to extract all the tables accessed by SCOTT, the following SQL would need to be supplied:

```
DTLDESCRIBE tables,SCOTT
```

DTLDESCRIBE FKEYS Column Information Retrieval

For example, to extract all the column information about the EMP table owned by SCOTT, the following SQL would need to be supplied: Notice the use of the comma to maintain field position after the reserved field.

```
DTLDESCRIBE columns,SCOTT,,EMP
```

DTLDESCRIBE FKEYS Primary Keys Information Retrieval

Example:

Table MFERNANDEZ.T4 has a primary key (PK4) defined on columns T41,T42.

```
DTLDESCRIBE pkeys,mfernandez,,t4
```

Result:

```
MFERNANDEZ|T4|T41|1|PK4|
MFERNANDEZ|T4|T42|2|PK4|
```

The numbers before PK4 are the order of the columns in the key definition.

DTLDESCRIBE FKEYS Foreign Keys Information Retrieval

Example 1:

Given a table, find all the foreign keys defined on that table.

Table MFERNANDEZ.T2 has two foreign keys.

(FK21) to T1 from column T21 to T11 with UPDATE rule RESTRICT and DELETE rule = DELETE

and

(FK23) to T3 from columns T22,T23 to T31,T32 with UPDATE rule NOACTION and DELETE rule = NOACTION

```
DTLDESCRIBE fkeys,,,mfernandez,,t2
```

Example 1 Result:

```
MFERNANDEZ|T1|T11|MFERNANDEZ|T2|T21|1|RESTRICT|RESTRICT|FK21|SQL010827171053710|
MFERNANDEZ|T3|T31|MFERNANDEZ|T2|T22|1|NO ACTION|NO ACTION|FK23|PK3|
MFERNANDEZ|T3|T32|MFERNANDEZ|T2|T23|2|NO ACTION|NO ACTION|FK23|PK3|
```

Example 2:

Given a "parent table", find all the foreign keys that point to that table

```
DTLDESCRIBE fkeys,mfernandez,,t1
```

Example 2 Result:

```
MFERNANDEZ|T1|T11|MFERNANDEZ|T2|T21|1|RESTRICT|RESTRICT|FK21|SQL010827171053710|
```

It is valid to query on both the parent and the foreign tables as shown below.

```
DTLDESCRIBE fkeys,mfernandez,,t*,mfernandez,,q*
```

DTLDESCRIBE SCHEMAS Qualifier

Enables you to list the available schemas:

```
dtldescribe schemas, [filter_pattern],,,,,,, [access_method]
```

After the basic syntax of `dtldescribe schemas`, the parameters in square brackets `[]` are optional. Except for the optional *filter_pattern* and *access_method*, all of the other positional parameters are ignored. If any parameters are coded, you must include the intervening commas so that the parser can determine which field is included.

The following table describes the parameters:

Parameter	Description
<i>filter_pattern</i>	Supply a pattern to filter the schemas that are returned to just those that start with the supplied character. Use an asterisk (*) as a wildcard to match one or more characters of any type.
<i>access_method</i>	<p>For NRDB,NRDB2:</p> <p>Specifies the valid access method or methods for which information is returned. The results of the DTLDESCRIBE are filtered to return information for only the specified access methods. Use this parameter to return a list of data maps that are applicable for certain access methods. Any number or combination of access method identifiers can be used, for example, OS or AKT. If no value is specified, the DTLDESCRIBE returns information for all access methods by default.</p> <ul style="list-style-type: none"> - ADABAS - A - CAPX / RT - C - DATACOM - X - DB2 - Z - DB2390IMG - G - DB2UNLD - W - DL1 - D - ESDS - E - IDMS - I - KSDS - K - MQ - Q - ODBA - O - RRDS - N - SEQ - S - TAPE - T - USER - U

For example:

```
DtlDescribe schemas, d* ,,,,,,,,, C
```

This example returns dev_schema but not eval_schema, if dev_schema is using the CAPX access method.

This function is implemented for CAPX, CAPXRT, DB2, DB2400C, IMSUNLD, MSSQL, NRDB, NRDB2, ORACLE, and DB2390IMG.

DTLDESCRIBE Qualifiers

DTLDESCRIBE provides qualifiers based on data source.

DTLDESCRIBE Qualifiers for DB2 for i5/OS Sources

The following table lists the DTLDESCRIBE qualifiers for DB2 for i5/OS sources:

Field Name	Explanation
Qualifier 1	Optional, Schema name
Qualifier 2	Optional, Table owner

DTLDESCRIBE Qualifiers for DB2 for z/OS Sources

The following table lists the DTLDESCRIBE qualifiers for DB2 for z/OS sources:

Field Name	Explanation
Qualifier 1	Optional, Creator
Qualifier 2	Optional, DBName

DTLDESCRIBE Qualifiers for DB2 for Linux, UNIX, and Windows Sources

The following table lists the DTLDESCRIBE qualifiers for DB2 for Linux, UNIX, and Windows sources:

Field Name	Explanation
Qualifier 1	Optional, Schema
Qualifier 2	Optional, Definer (Creator ID)

DTLDESCRIBE Qualifiers for NRDB Sources

The following table lists the DTLDESCRIBE qualifiers for nonrelational data sources such as IMS, VSAM, IDMS, Adabas, Datacom, and sequential files:

Field Name	Explanation
Qualifier 1	Optional, gives the first token of the data map name
Qualifier 2	Optional, gives the second token of the data map name

DTLDESCRIBE Qualifiers for NRDB2 Sources

The following table lists the DTLDESCRIBE qualifiers for nonrelational two-tier data formats:

Field Name	Explanation
Qualifier 1	Optional, gives the first token of the data map name
Qualifier 2	Not used

DTLDESCRIBE Qualifiers for Oracle Sources

The following table lists the DTLDESCRIBE qualifiers for Oracle sources:

Field Name	Explanation
Qualifier 1	Optional, Schema name
Qualifier 2	Not used

Restrictions on PowerExchange Nonrelational SQL

PowerExchange places certain restrictions on the SQL that it supports to access nonrelational data sources.

Reordering or aggregating the result set is not supported. The sequence of data in a result set depends on the physical sequence and whether data was accessed via an index.

The following table lists and describes SQL expressions that are not supported:

Unsupported SQL Expression or Keyword	Comment
CREATE	Creation of nonrelational tables and indexes is not supported. Define these objects by using the underlying nonrelational system.
GROUP BY	Aggregation is not supported.
HAVING	Aggregation is not supported.
JOIN	SQL on more than one table is not supported.
ORDER BY	Reordering result sets is not supported.
SELECT * FROM TABLE1 WHERE COL1 IN SELECT COL2 FROM TABLE2	Subselects are not supported.
SELECT COUNT(*)	Aggregation is not supported.
UNION	SQL on more than one table is not supported.

Truncating a Target Table

For nonrelational target types, PowerExchange and PowerCenter support clearing target tables before writing to them in a PowerCenter workflow.

You can clear a target table in the following ways:

- Include SQL to truncate the table in the **Pre SQL** session attribute for the target table.
- Select the **Truncate target option** session property for a VSAM target that uses a PWXPC connection.
- Select the **Truncate Target Table** property for a nonrelational target that uses an PowerExchange ODBC connection.

The first two tokens in the SQL can be either TRUNCATE TABLE or DELETE FROM. The final tokens in the SQL define the fully qualified table from which the data map can be determined. For example, you could use one of the following statements:

```
TRUNCATE TABLE schema.map_table  
DELETE FROM schema.map_table
```

If you select the **Truncate target option** session property or the **Truncate Target Table** ODBC property, PowerExchange generates a statement, which you cannot modify, in one of these forms.

The system loads the data map for the table and determines the underlying map type.

For map types of ESDS, KSDS, RRDS, SEQ, or TAPE, the file defined in the data map (or overridden at run time) is opened in write mode and closed so that existing contents of the file are cleared. This method of clearing files is useful because otherwise, if no data is processed in a workflow, the output file might contain records from an earlier run that are retained. PowerExchange defers opening of output files until actual data is being written.

The entire file is always cleared. In the case of multiple-record files, any of the table names can be chosen and data for all tables is cleared. Although it is physically possible for an individual table to be cleared from KSDS and RRDS files, you cannot achieve this result by defining TRUNCATE TABLE or DELETE FROM in the **Pre SQL** session property. It might be possible to clear part of a KSDS file using DELETE SQL with a WHERE clause that is run as part of a workflow, but not by defining the statement in the **Pre SQL** session property.

Clearing a target table by using one of these methods is not supported for other data map types.

Note: VSAM data sets must be defined with the REUSE option for table truncation to function correctly.

Inserting Rows into a Table

A PowerCenter workflow can write to NRDB target types ADABAS, IMS, SEQ, TAPE, and VSAM. For these target types, it is possible to execute INSERT SQL to insert rows with hard-coded data values into the output table.

CHAPTER 10

PowerExchange Globalization

This chapter includes the following topics:

- [PowerExchange Globalization Overview, 346](#)
- [Default Code Pages Installed with PowerExchange, 347](#)
- [General Code Pages Available with PowerExchange, 347](#)
- [Source-Specific Code Pages, 348](#)
- [User-Defined Code Pages, 355](#)
- [ICUCHECK Utility, 360](#)
- [Code Pages Shipped with PowerExchange, 367](#)
- [Handling Conversion Errors and Special Situations, 379](#)
- [Type 1 Fallback Mappings and Multibyte Data, 381](#)

PowerExchange Globalization Overview

PowerExchange supplies default code pages based on International Components for Unicode (ICU) open source software.

PowerExchange uses the following type of code page specification:

- **Default Code Pages.** During PowerExchange installation, default code pages are installed in the installation directory for each PowerExchange system.
- **General Code Pages.** PowerExchange uses the CODEPAGE statement in the DBMOVER configuration file to define the general code pages for transmitting or receiving data or metadata between systems.
- **Code Pages for Data Sources.** Where possible, PowerExchange derives the code page of data from the database metadata. PowerExchange can use data source-specific configuration parameters to define the code pages for transmitting or receiving data or metadata between data sources.
- **User-Defined Code Pages.** Where it is not possible to derive the required code pages, or where data is loaded in a different code page from that which the database metadata expects, you must explicitly define the code pages for the databases that you use.

When the PowerExchange Listener starts up, it reports the code pages being used.

Default Code Pages Installed with PowerExchange

During PowerExchange installation, default code pages are installed in the installation directory for each PowerExchange system.

The following table lists the default code page for each system:

System	Code Page
i5/OS	IBM-037
UNIX and Linux	ISO-8859
Windows	ISO-8859
z/OS	IBM-037

If these code pages are sufficient for your PowerExchange installation, do not configure any additional code pages. Typically, the default code pages are sufficient in the following circumstances:

- All data is held in single-byte code pages.
- Few accented characters are used.
- Data on databases and legacy files are in the same code page.

If the default code pages are not sufficient for your PowerExchange installation, use the CODEPAGE statement in the DBMOVER configuration file to specify general code pages.

General Code Pages Available with PowerExchange

Use general code pages if the default code pages that PowerExchange supplies are not sufficient for your installation. PowerExchange uses general code pages to transmit data or metadata between systems.

Typically, general codes pages are used in the following circumstances:

- Multibyte data is processed.
- Data is outside of the ISO-8859 range on Linux, UNIX, or Windows, or outside of the IBM-037 range on i5/OS or z/OS.

Enter general code pages in the CODEPAGE statement of the DBMOVER configuration file. You can define the following types of general code pages in the CODEPAGE parameter:

- Control code page
- Data control code page
- SQL code page

The following table describes the uses of each code page type that can be specified in the CODEPAGE parameter:

Code Page	Uses
Control code page	<ul style="list-style-type: none">- Names of databases, tables columns, or files.- User IDs or passwords.- Error messages.
Data control code page	<ul style="list-style-type: none">- Column and parameter data used on nonrelational database types where the code pages have not been specified, for example, at the field or data map level. Parameter data refers to data sent to the server and substituted where parameter markers (question marks) are present in Delete, Select, or Update SQL.
SQL code page	<ul style="list-style-type: none">- Code page of SQL used to access data. Typically, the SQL code page is the same as the control code page because table names can be processed by both. The SQL code page must be sufficiently precise to handle any literals.

If you specify the control code page and omit the data control and SQL code pages, PowerExchange uses the control code page for the data control and SQL code pages. For example, the following statements are equivalent:

```
CODEPAGE=(UTF8)
CODEPAGE=(UTF8,UTF8,UTF8)
```

If you omit the CODEPAGE parameter on Linux, UNIX, or Windows, PowerExchange uses the following default values:

```
CODEPAGE=(ISO-8859,ISO-8859,ISO-8859)
```

If you omit the CODEPAGE parameter on i5/OS or z/OS, PowerExchange uses the following default values:

```
CODEPAGE=(IBM-037,IBM-037,IBM-037)
```

Note: The PowerExchange Navigator overrides the code page specified in the DBMOVER configuration file, so that it can handle all characters. The PowerExchange Navigator uses UTF8 for the control, data, and SQL code pages.

Source-Specific Code Pages

Review the topics for your data source to learn about the source-specific code page parameters that you can set and how PowerExchange derives the internal code page numbers.

How PowerExchange Determines Internal Code Page Numbers by Data Source

PowerExchange uses internal code page numbers to uniquely identify code pages.

PowerExchange uses the internal code page numbers in many contexts, such as the following situations:

- Exchanging control, data, and SQL code pages between the PowerExchange Listener and a calling application when processing open requests that are sent across the network.

- Performing a SQL DTLDESCRIBE operation that processes CHAR and VARCHAR columns when importing metadata.
- Processing an NRDB data map that defines code pages for fields and the entire map.

You can use the ICUCHECK utility to list the defined code page numbers.

The following table describes how the default internal code page numbers are derived by source or target type:

Source or Target Type	How Code Page Numbers Are Derived
DB2 on i5/OS	PowerExchange determines the internal code page number from the CCSID of the column and code page aliases. Optionally, columns that have no CCSIDs can be mapped to CHAR columns with code pages using the optional the DB2_BIN_CODEPAGE and DB2_BIN_AS_CHAR configuration statements. Otherwise, these columns are mapped to BIN columns and can only be processed in hexadecimal format.
DB2 on z/OS bulk data movement	For each source or target DB2 subsystem, PowerExchange determines the internal code page numbers for data columns based on the column CCSIDs and the DB2CODEPAGE statement in the DBMOVER configuration file. <ul style="list-style-type: none"> - For single-byte columns, the internal code page number is based on the CCSID of the column and the first <i>sbcscsids</i> value in the EBCDIC_CCSID or PLAN_CCSID parameter of the DB2CODEPAGE statement. - For graphic double-byte columns, the internal code page number is based on the CCSID of the column and the <i>graphic_ccsid</i> value in the EBCDIC_CCSID or PLAN_CCSID parameter of the DB2CODEPAGE statement. - For mixed-byte columns, the internal code page number is based on the CCSID of the column and the <i>mixed_ccsid</i> value in the EBCDIC_CCSID or PLAN_CCSID parameter of the DB2CODEPAGE statement.
Nonrelational bulk data movement	PowerExchange determines the internal code page number in the following order: <ol style="list-style-type: none"> 1. The code page of the field from which the column was derived. This code page and field are specified in the data map. 2. The code page of the data map. 3. The CODEPAGE parameter for the data control code page, on the server where the NRDB access method runs.
Nonrelational CDC	PowerExchange determines the internal code page number in the following order: <ol style="list-style-type: none"> 1. The code page of the field from which the column was derived. This code page and field are specified in the data map. 2. The code page of the data map. 3. The CODEPAGE parameter for the data control code page, on the server where the NRDB access method runs. PowerExchange records the code page of a field or data map in the CCT file when you create a capture registration.
Oracle bulk data movement	PowerExchange determines the internal code page number from the character set portion of the NLS_LANG environment variable. Optionally, you can use the ORACLECODEPAGE parameter in the DBMOVER configuration file.

Source or Target Type	How Code Page Numbers Are Derived
Oracle CDC	The PowerExchange internal code page for the columns from which changes are captured is always UTF-8.
Microsoft SQL Server	PowerExchange determines the internal code page number from the collation sequence of the database.

DB2 for i5/OS Code Page Parameters

Some systems define the data in tables as having a CCSID of 65535 (X'FFFF'). By default, PowerExchange treats such data as binary, which prevents the data from being translated. Binary data can be difficult to load onto a target system.

Optionally, set the following parameters in the DBMOVER configuration file to specify code page values for DB2 for i5/OS:

- DB2_BIN_AS_CHAR
- DB2_BIN_CODEPAGE

DB2_BIN_AS_CHAR Statement

Use the DB2_BIN_AS_CHAR statement to specify whether binary data is treated as character data.

```
DB2_BIN_AS_CHAR={Y|N}
```

DB2_BIN_CODEPAGE Statement

Use the DB2_BIN_CODEPAGE statement to associate the required single- and double-byte CCSIDs.

```
DB2_BIN_CODEPAGE=(SBCS_CCSID,DBCS_CCSID)
```

The following table describes the parameters in the DB2_BIN_CODEPAGE statement:

Parameter	Description
SBCS_CCSID	Specifies the single-byte character set.
DBCS_CCSID	Specifies the double-byte character set.

Note: PowerExchange supports ICU code pages only on DB2 for i5/OS 5.1 and later. For DB2 for i5/OS 4.5, PowerExchange uses static code pages and cannot access any column for which the CCSID maps to an ICU code page.

Use the DB2 access method, not the DB2400C access method, for code page support on DB2 for i5/OS.

DB2 for z/OS Code Page Parameters

You can use the DB2CODEPAGE statement in the DBMOVER configuration file to specify single-byte, graphic double-byte, and mixed-byte code pages for CHAR and VARCHAR columns in a DB2 for z/OS source or target subsystem.

The DB2CODEPAGE statement applies to bulk data movement operations only. For DB2 CDC, PowerExchange always uses the native code page of the data.

During PowerExchange installation, a bind job binds the DB2 plan for PowerExchange as EBCDIC, without specifying the CCSIDs. As a result, PowerExchange uses the default application programming CCSIDs that were defined for the DB2 subsystem when it was created. PowerExchange retrieves these CCSIDs from the DB2 catalog tables and uses them along with the DB2CODEPAGE parameters to determine the code page to use.

Verify that the CCSIDs specified in the DB2CODEPAGE parameters match the CCSIDs that are specified for the DB2 subsystem on the DSNTIPF panel and in the DB2 installation job for the DSNHDECP load module.

RELATED TOPICS:

- [“DB2CODEPAGE Statement” on page 106](#)

DB2CODEPAGE Statement Usage

The DB2CODEPAGE statement defines CCSIDs that PowerExchange uses to process data in CHAR and VARCHAR columns during bulk data movement. Each DB2CODEPAGE statement applies to a specific source or target DB2 subsystem.

The DB2CODEPAGE statement has the following syntax:

```
DB2CODEPAGE=(db2_subsystem_ssid
              [,DB2TRANS={P|N|R}]
              [,MIXED={N|Y}]
              [,EBCDIC_CCSID=({sbcscsid|037}
                              ,{graphic_ccsid|037}
                              ,{mixed_ccsid|037})]
              [,ASCII_CCSID=({sbcscsid|850}
                              ,{graphic_ccsid|65534}
                              ,{mixed_ccsid|65534})]
              [,UNICODE_CCSID=({sbcscsid|367}
                                ,{graphic_ccsid|1200}
                                ,{mixed_ccsid|1208})]
              [,PLAN_CCSID=(sbcscsid
                            ,graphic_ccsid
                            ,mixed_ccsid)]
              [,REMAPn=(current_data_ccsid),(remapped_data_ccsid)
              )
```

You must include the DB2 subsystem SSID. If DB2TRANS=P, you must also include the EBCDIC_CCSID parameter. The other parameters are optional and depend on the code page conversion requirements in your PowerExchange environment. For more information, see [“DB2CODEPAGE Statement” on page 106](#).

For each _CCSID parameter, specify a value for the following types of CCSIDs:

- A SCCSID for single-byte data.
- A GCCSID for graphic data.
- A MCCSID for mixed data that contains both single-byte and double-byte characters.

A value of 65534 indicates no CCSID.

You can define a maximum of 24 DB2CODEPAGE statements in one DBMOVER configuration file for a single PowerExchange Listener.

If PowerExchange connects to a DB2 subsystem that does not have a corresponding DB2CODEPAGE statement, the data is processed with the default code page of the PowerExchange Listener. The PowerExchange Listener reports the code page that is being used.

If data is stored in more than one code page on DB2 or on other file systems that the PowerExchange Listener accesses, you must specify a DB2CODEPAGE statement for each DB2 subsystem that PowerExchange accesses.

For more information about DB2 CCSIDs, see the *IBM DB2 for z/OS Installation Guide*.

DB2CODEPAGE Statement Examples

Review these examples to learn how to define and use the parameters in the DB2CODEPAGE statement of the DBMOVER configuration file. The DB2CODEPAGE statement is used for DB2 for z/OS bulk data movement sources and targets.

Example 1. Look Up the Code Pages Set for the DB2 DSNHDECP Module

Look up the EBCDIC code page values for Japanese Extended Katakana in the DB2 installation job for the DSNHDECP module. Then configure the DB2CODEPAGE statement to match these code page values.

The installation job contains the following EBCDIC CCSID values for the DB2 subsystem:

```
000273 //SYSIN      DD  *
000274      DSNHDECM    CHARSET=ALPHANUM,
000275                      ASCCSID=1041,                      X
000276                      AMCCSID=942,                      X
000277                      AGCCSID=301,                      X
000278                      SCCSID=290,                      X
000279                      MCCSID=930,                      X
000280                      GCCSID=300,                      X
000281                      USCCSID=367,                      X
000282                      UMCCSID=1208,                     X
000283                      UGCCSID=1200,                     X
000302                      END
000303 //*
```

Use these CCSID values in the following PowerExchange DB2CODEPAGE parameters:

```
DB2CODEPAGE=(D91G
              ,EBCDIC_CCSID=(290,300,930)
              ,ASCII_CCSID=(1041,301,942)
              ,UNICODE_CCSID=(367,1200,1208)
            )
```

Example 2: Use the Default DB2CODEPAGE Configuration

By default, DB2 translates single-byte CCSIDs for CHAR and VARCHAR character strings into the equivalent EBCDIC code pages in which the DB2 plan for PowerExchange was bound. If your DB2 subsystem contains no graphic or mixed data, you can use the default configuration.

In this case, PowerExchange uses the following DB2CODEPAGE parameters:

```
DB2CODEPAGE=(D91G
              ,DB2TRANS=P
              ,MIXED=N
              ,EBCDIC_CCSID=(037,037,037)
              ,PLAN_CCSID=(037,65534,65534)
              ,ASCII_CCSID=()
              ,UNICODE_CCSID=(367,1200,1208)
            )
```

Note that this statement is equivalent to the following minimal configuration:

```
DB2CODEPAGE=(D91G
              ,DB2TRANS=P
              ,MIXED=N
              ,EBCDIC_CCSID=(037,037,037)
              ,UNICODE_CCSID=(367,1200,1208)
            )
```

If you omit the DB2CODEPAGE statement from the DBMOVER file, PowerExchange automatically generates these code page parameter values for the DB2 subsystem.

The EBCDIC_CCSID parameter specifies a CCSID for single-byte data only. No code page is specified for graphic and mixed data, as indicated by the value 65534. Because the DB2 catalog tables store data with Unicode encoding, this default configuration also includes the UNICODE_CCSID parameter.

The EBCDIC_CCSID values must be consistent with the code pages defined for the DB2 plan for PowerExchange and with the default code pages defined in the DB2 installation job for the DB2 DSNHDECM load module.

Example 3. Redirect the Default EBCDIC Code Pages to the DB2 Plan Code Pages

The EBCDIC_CCSID code pages must match the code pages that were defined for the DB2 DSNHDECP load module when the DB2 subsystem was created. If these code pages are different from the code pages in which the DB2 plan for PowerExchange was bound, you can use the PLAN_CCSID parameter to direct DB2 to use the plan code pages.

Use the PLAN_CCSID parameter in the following situations:

- The EBCDIC_CCSID code pages do not have an ICU conversion table that PowerExchange can use for ICU-based code page conversion.
- The EBCDIC_CCSID code pages match the default code pages that were defined for the DB2 subsystem but differ from the EBCDIC code pages of the source or target table.

For example, specify the following DB2CODEPAGE parameters:

```
DB2CODEPAGE=(D91G
              ,DB2TRANS=P
              ,MIXED=N
              ,EBCDIC_CCSID=(290,300,930)
              ,PLAN_CCSID=(037,309,309)
              )
```

The PLAN_CCSID values override the EBCDIC_CCSID values.

Example 4: Map ASCII and UNICODE Code Pages to EBCDIC Code Pages

The DB2 subsystem "D91G" contains data with ASCII or Unicode encoding. You want DB2 to translate the ASCII and Unicode code pages of the data into the equivalent EBCDIC code pages that were defined in the DB2 plan that was bound for PowerExchange.

In this case, specify the following parameters in the DB2CODEPAGE statement:

```
DB2CODEPAGE=(D91G
              ,DB2TRANS=P
              ,MIXED=N
              ,EBCDIC_CCSID=(290,300,930)
              ,ASCII_CCSID=(1041,301,942)
              ,UNICODE_CCSID=(367,1200,1208)
              )
```

Because DB2TRANS=P, DB2 translates the code pages. DB2 redirects the ASCII_CCSID and UNICODE_CCSID code pages to the EBCDIC_CCSID code pages.

Example 5. Use the Native Code Pages of the Data for Columns with Mixed Data

The DB2 subsystem "D91G" contains CHAR FOR MIXED DATA columns that use ASCII or Unicode encoding. These columns can contain a mix of single-byte and double-byte characters.

When DB2 translates mixed data to an EBCDIC equivalent CCSID, DB2 uses the shift-in and shift-out characters (X'OE' and X'OF') to change between the single-byte and double-byte characters. As a result, the data might become too long to fit in some columns. In this situation, you can prevent DB2 from translating the data to the EBCDIC equivalent code page and use the native ASCII or Unicode code page in which the data is stored instead.

Specify the following parameters in the DB2CODEPAGE statement:

```
DB2CODEPAGE=(D91G
              ,DB2TRANS=N
              ,MIXED=Y
              )
```

The DB2TRANS=N parameter prevents DB2 from performing code page translation. The MIXED=Y parameter indicates that the column character strings can contain mixed data.

Note: No *_CCSID parameters are required because DB2 performs no code page translation.

Example 6. Direct DB2 to Remap Code Pages That Do Not Have an ICU Conversion Table

If an ICU conversion table is not available for an ASCII or Unicode CCSID and you cannot remap the CCSID to an EBCDIC CCSID in which the DB2 plan for PowerExchange is bound, DB2 can remap the CCSID to the native CCSID of the column data. You must include at least one REMAPn parameter to identify the CCSID to remap and the data CCSID to use. You can define up to six REMAPn parameters, one per table.

In this example, DB2 remaps the Japanese CCSID 301 to the corresponding Unicode double-byte CCSID 1200, or UTF-16. An ICU conversion table is not available for CCSID 301 but is available for CCSID 1200.

Specify the following parameters in the DB2CODEPAGE statement:

```
DB2CODEPAGE=(D91G
              ,DB2TRANS=R
              ,REMAP1=(301,1200)
            )
```

Example 7. Direct PowerExchange to Remap Incorrect Code Pages

In certain circumstances, you might need PowerExchange to remap an incorrect code page without any DB2 translation. For example, this type of remapping might be necessary in the following circumstances:

- The DB2 data is not in the code page that DB2 reports based on the DB2 catalog, possibly because the data was loaded incorrectly.
- The current code page does not correctly translate certain characters in the data, such as square brackets or the Euro currency symbol.

In this case, use the DB2TRANS=N parameter and REMAPn parameter to specify the correct code page. PowerExchange substitutes the correct code page for the incorrect one.

Specify the following parameters in the DBMOVER file:

```
DB2CODEPAGE=(D91G
              ,DB2TRANS=N
              ,REMAP1=(incorrect_cp,correct_cp)
            )
```

Oracle Code Page Parameters

Oracle passes character data to PowerExchange based on the character set portion of the NLS_LANG environment variable. You must define the NLS_LANG environment variable. You might also need to specify an ORACLECODEPAGE statement in the dbmover.cfg configuration file.

- If the NLS_LANG environment variable specifies UTF8 or AL32UTF8, PowerExchange determines the code page with which to process connection strings, SQL statements, column data, and parameter data.
- If the NLS_LANG environment variable specifies a character set other than UTF8 or AL32UTF8, you must define the ORACLECODEPAGE statement in the dbmover.cfg configuration file.

The ORACLECODEPAGE statement determines the code pages that PowerExchange and PowerCenter use for an Oracle database during bulk data movement operations. These code pages must match the character set specified in the NLS_LANG environment variable.

Use the following syntax to enter this statement:

```
ORACLECODEPAGE=(tnsname_host,pwx_codepage,pc_codepage)
```

The statement contains the following positional parameters:

tnsname_host

Required. An entry in the Oracle tnsnames.ora configuration file for an Oracle database. The entry defines the database address that PowerExchange uses to connect to the database.

pwx_codepage

Required. A code page number or alias name that PowerExchange uses to identify a code page. To determine valid PowerExchange code page and alias values, use the ICUCHECK utility to generate report 5, "PowerExchange Code Page Names and Aliases."

Note: PowerExchange supports some common Oracle character set names as aliases to code pages.

pc_codepage

Optional. A name that controls the processing of the SQL statements that PowerExchange passes to Oracle on behalf of PowerCenter bulk data movement sessions. PowerExchange supplies a default that is usually suitable.

Do not specify this parameter except in special circumstances when the default does not work. For example, if you specify a user-defined ICU code page for the *pwx_codepage* parameter, you need to specify this parameter.

For example, if the NLS_LANG environment variable specifies Korean_Korea.KO16MSWIN949, define the following ORACLECODEPAGE statement in the dbmover.cfg file:

```
ORACLECODEPAGE=(KO102DTL,MS949)
```

You can specify up to 20 ORACLECODEPAGE statements, each for a separate database, in a dbmover.cfg file.

If PowerExchange uses the same NLS_LANG environment variable to access multiple Oracle databases, you do not need to specify a separate ORACLECODEPAGE statement for each database. Instead, specify a single ORACLECODEPAGE statement and leave the *tnsname_host* parameter blank. The specified code page then applies to all databases that have an entry in the tnsnames.ora file. The following example shows a statement without a *tnsname_host* parameter:

```
ORACLECODEPAGE=(,MS1252)
```

If you enter an incorrect PowerCenter code page value, the ODLNumResultCols routine usually reports Oracle return code 911.

RELATED TOPICS:

- ["ICUCHECK Utility" on page 360](#)

User-Defined Code Pages

PowerExchange can use the following types of user-defined code pages:

- **ICU-compatible code pages.** These code pages are binary CNV files that are created from source definitions in a UCM file when you run the ICU makeconv utility. ICU supports all types of code pages including multibyte code pages.
- **PowerExchange static code pages.** These code pages are defined as a 16-by-16 matrix in text files. PowerExchange static code pages are restricted to single-byte code pages. Data cannot be converted to ICU code pages.

Note: If you are upgrading PowerExchange from a release earlier than 9.0 to 9.0 or later, you must re-create the CNV files on each platform. To re-create the files, run the V900 or later version of the ICU makeconv utility.

Adding User-Defined ICU-Compatible Code Pages

Use the following task flow to add custom ICU code pages into PowerExchange code page internal slots:

1. Create the Unicode mapping (UCM) file, which defines the mapping between Unicode characters and the code page characters.
2. Run the ICU makeconv utility to build the run time CNV files on the systems where PowerExchange uses the code pages.
3. Use the PowerExchange DBMOVER configuration statements ICUCONVERTER, ICUALIAS, and ICUDATADIR to add the new code page into an internal slot.
4. In the PowerExchange Navigator, run a database row test to view the data.

Creating the UCM File

The UCM file maps the relationship between Unicode characters and hexadecimal values in the code page.

Edit the UCM file in a text editor. The file must contain only 7-bit ASCII characters on Linux, UNIX, and Windows systems, or the equivalent characters on EBCDIC systems.

The following table shows valid values for the precision indicator:

Precision Indicator	Description
0	Normal, round-trip mapping. A character retains the same hexadecimal value after being copied from the code page to Unicode and then copied back to the code page again.
1	Fallback mapping. A character is copied from Unicode to the code page but not back to Unicode again. If a character was copied back to Unicode again, it would get a new Unicode value.
2	Invalid character. The character will be replaced by the substitution character.
3	Reverse-fallback mapping. A character is copied from the code page to Unicode but not back to the code page. If the character was copied back to the code page again, it would get a different hexadecimal value.

Most characters are defined as round-trip mappings and have Unicode or hexadecimal values.

Sometimes, however, it is necessary to use one-way mappings, which cause an overlap between characters. For example, if a character is in the source code page but not in the target code page, or if a character is invalid, you can define a reverse-fallback mapping for it. When PowerExchange reads the data, the reverse-fallback mapping causes the character to be changed to a Unicode character that is a near match.

For more information about UCM file format, see the following Web site:

<http://icu.sourceforge.net/userguide/conversion-data.html#ucmformat>.

Running the ICU makeconv Utility

The makeconv utility converts a source code page definition from a UCM file into a binary CNV file.

The makeconv utility is an open source program that is available from the ICU Web site.

The makeconv utility embeds the ICU version number into the CNV file name.

If you created custom code pages with earlier versions of ICU, you must regenerate the CNV files using the V900 version of the makeconv utility.

Use the following syntax to run the makeconv utility and build a binary CNV file for the code page xxxx:

```
makeconv -p ICUDATA xxxx.UCM
```

When you run the V900 version of the makeconv utility, The generated .cnv file names are prefixed with the letters PM. For example, if you run the makeconv utility for the code page xxxx, the generated file is named PMxxxx.CNV.

You can use the makeconv utility on the following systems:

- i5/OS. On i5/OS, the makeconv utility runs as a *PGM from a SBMJOB command.
- UNIX and Linux.
- Windows.
- z/OS. On z/OS, the makeconv utility runs within the HFS UNIX system.

For more information, see the ICU makeconv utility documentation at the following Web site:

<http://userguide.icu-project.org/icudata>.

Adding a Custom ICU Code Page

PowerExchange defines code pages internally using Code Page Numbers (CPNs). You can use the ICUCHECK utility to list the defined CPNs.

User-defined ICU code pages must be defined in slots 301 to 340.

Add a custom code page on each system that is involved in PowerExchange change data capture or bulk data movement. For example, if you are moving data from a data source on z/OS to a target on UNIX, you must add a code page definition to the DBMOVER configuration file on the z/OS system and on the UNIX system.

You must also add the new code page to the dbmover.cfg file on Windows so that you can view the data from the PowerExchange Navigator when performing a row test.

To add a custom ICU code page:

1. Move the CNV file to the required directories on each system.

The following table lists where CNV files are located for each system:

System	Location of CNV Files
i5/OS	UNIX style directory. The makeconv utility is run from the QSH shell. Set ICUDATADIR to the directory name.
UNIX and Linux	CNV files are located in the PowerExchange installation directory. If PowerCenter workflows are run on the machine, you must copy the CNV files to the server/bin directory which also contains the RES files.

System	Location of CNV Files
Windows	CNV files are located in the PowerExchange installation directory. If PowerCenter workflows are run on the machine, you must copy the CNV files to the server/bin directory which also contains the RES files.
z/OS	UNIX style directory. The makeconv utility is run from USS. Set ICUDATADIR to the directory name.

- Run ICUCHECK to determine which code pages are defined and to verify that the current configuration and dynamic link libraries are correct. Add the ICUDATADIR statement to the DBMOVER configuration file. In this file, specify the location of the CNV file.
- Add an ICUCONVERTER(*cpn*,ADD...) statement to the DBMOVER configuration file. Typically, the first user-defined ICU code page uses CPN 301, the next one uses 302, and so on.
- If the code page is used by DB2 to map a CCSID to a code page, add an ICUALIAS=(*cpn*,ADD...) statement to the DBMOVER configuration file.
- If the code page replaces an existing code page that is used by DB2 to map a CCSID to a code page, add an ICUALIAS=(*cpn*,DELETE...) statement to the DBMOVER configuration file.
Consult the ICUCHECK reports to determine the CPN of the existing code page.
- Run ICUCHECK again to verify that the syntax of the PowerExchange configuration statements is correct and that the CNV file can be loaded. ICUCHECK does not start if the parameters contain syntax errors.

RELATED TOPICS:

- [“ICUCHECK Utility” on page 360](#)

Viewing Data from the PowerExchange Navigator

If the database type is nonrelational, define the code page at either the data map level for all CHAR and VARCHAR columns, or at the field level.

If the database type is DB2, verify that the CPxxx alias points to the new code page in the assigned CPN, such as 301, and delete existing aliases that are pointing at a different code page.

To view data from the PowerExchange Navigator:

- In the **Database Row Test** dialog box, select **Columns in Fetch** and **Extensions**.
- Perform a database row test.
- Verify that the internal CPN number is the number that you assigned, such as 301.
- In the **Database Row Test** dialog box, select **Data in Fetch**.
- Run a row test of the data. The data can be accessed through Personal Metadata or an NRDB data map.
- Verify that the data is as expected.

PowerExchange Static Code Pages

You can define up to 10 external code page tables using fixed names. Code pages are defined in sequential text files that PowerExchange reads at run time. The parameter is specified in the following format:

```
CODEPAGE=( , USRCPNN, )
```

The variable *NN* is a number from 00 to 09, such as USRCP07.

Add a PowerExchange static code pages in the following manner:

- Add a control, data, or SQL code page in the DBMOVER configuration file.
- Add field or data map-level code pages in data maps for nonrelational database sources from the PowerExchange Navigator.

The following table specifies the code page file name and location that you must use on each system:

System	Location and File name
i5/OS	USRCPnn member that is specified in the CODEPAGE parameter of the DBMOVER configuration file.
UNIX and Linux	USRCPnn.cp file in the directory where PowerExchange is installed.
Windows	USRCPnn.cp file in the directory where PowerExchange is installed.
z/OS	Data set that is allocated by the USRCPnn DD in the PowerExchange Listener or Client JCL.

You must specify 256 hexadecimal characters in 16 rows, each with 16 characters. Every character-pair of hexadecimal digits must be separated by a blank. The following example shows the code page table of an external code page file:

```

00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
40 4F 7F 7B 5B 6C 50 7D 4D 5D 5C 4E 6B 60 4B 61
F0 F1 F2 F3 F4 F5 F6 F7 F8 F9 7A 5E 4C 7E 6E 6F
AC C1 C2 C3 C4 C5 C6 C7 C8 C9 D1 D2 D3 D4 D5 D6
D7 D8 D9 E2 E3 E4 E5 E6 E7 E8 E9 AE BE 9E EC 6D
8C 81 82 83 84 85 86 87 88 89 91 92 93 94 95 96
97 98 99 A2 A3 A4 A5 A6 A7 A8 A9 8E BB 9C CC 00
00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00 AA B0 B1 9F B2 6A B5 BD B4 9A 8A BA CA AF BC
90 8F EA FA E0 A0 B6 B3 9D DA 9B 8B B7 B8 B9 AB
64 65 62 66 63 67 5A 68 74 71 72 73 78 75 76 77
7C 69 ED EE EB EF 5F BF 80 FD FE FB FC AD 4A 59
44 45 42 46 43 47 D0 48 54 51 52 53 58 55 56 57
79 49 CD CE CB CF A1 E1 70 DD DE DB DC 8D C0 DF

```

Determining How the External Code Page File Is Specified

You can determine how the external code page file is specified.

To determine how the external code page file is specified:

1. Select a character in your local character set.
2. For the selected character, locate the equivalent character in the code page ISO-8859 and note the corresponding hexadecimal value.
3. Use this hexadecimal value to calculate the correct offset into the code page file that you are building.
4. At this offset, store the hexadecimal representation of your character.

For example:

1. Select 1 in your local z/OS character set. The character 1 is x'F1' in hexadecimal format.
2. Locate 1 in the code page ISO-8859 and find its corresponding hexadecimal value. This value is 31.
3. Locate offset x'31' in the code page file that is being defined. This offset location is at line 4, character 2, where the offset is relative to 1.

4. At that location, enter F1.

Because no mapping to Unicode exists, you cannot use any static code pages that you defined to perform the following tasks:

- View data by performing a row test in the PowerExchange Navigator.
- Convert characters to a target ICU code page.

ICUCHECK Utility

ICUCHECK lists all of the code pages and aliases that are defined for a PowerExchange installation. The program checks the contents of the PowerExchange code page control table against the ICU data library and lists any differences.

Use ICUCHECK to perform the following tasks:

- List the contents of the code page control table.
- Verify that the correct version of the ICU data library is used, and that the path is correct.
- Verify that every ICU code page in the PowerExchange code page control table can be loaded, either from the ICU data library, or from a CNV file in the ICU data directory.
- Find the spelling of ICU converter names so that you can copy and paste them into other areas, such as ODBC definitions on UNIX.
- Verify that the aliases are available for DB2 CCSID processing.

The output file contains reports that provide information about the available code pages. ICU provides the following reports:

- Report 1. Power Exchange Control Table for Simple Code Pages
- Report 2. PowerExchange Control Table for ICU Code Pages
- Report 3. Comparing Name and Character Size Information with ICU
- Report 4. Comparing ICU Code Page Information with PowerExchange
- Report 5. PowerExchange Code Page Names and Aliases
- Report 6. PMlocale Code Pages
- Report 7. Control Counts
- Report 8. Errors

ICUCHECK Syntax

Use the following program to display information on the screen:

```
icucheck.exe
```

On Linux, UNIX, or Windows, use the following syntax to run the ICUCHECK program and write the results to a text file:

```
icucheck.exe > icucheck.txt
```

On i5/OS, use the SBMJOB command to run the ICUCHECK program. The reports are written to a QPRINT file.

On z/OS, add statements for running the ICUCHECK program to the JCL for a job. The reports are written to a SYSPRINT file.

Report 1. Power Exchange Control Table for Simple Code Pages

This report displays a list of all simple code pages. Simple code pages are static single-byte code pages that provide a fast translation facility with the following limitations:

- You can translate data between static code pages, but you cannot translate between static and ICU code pages.
- You can use only a one-to-one mapping between characters, however you cannot use a one-to-many or many-to-one mapping between characters.

Static single-byte code pages are defined in the range of 1 to 40. User-defined static code pages are defined in the range of 31 to 40.

When translating between code pages, such as from CPN 1 "ISO-8859" to CPN 3 "IBM-037," the operating system builds a translation table of 256 bytes that contains the appropriate target values. To translate a particular value such as x'31', the system finds the value x'31' in the translation table and uses the translated value that is stored for it, x'F1'.

On systems that do not support ICU processing, static code pages provide the facility to translate between code pages. Each static code page must have an equivalent ICU code page that is used if the other code page is ICU. For example, if you move data from CPN 3 "IBM-37" to CPN 41 "UTF-8," the system converts from CPN 183 "ibm-37_P100-1995" to CPN 41 "UTF-8."

The following report is a sample PowerExchange control table for simple code pages report:

```
Report 1 : Power Exchange control table for Simple code pages
=====

NUM_STATIC_CODEPAGES = 41

CPN Name      PMlocale      PowerCenter name      Space CPN2 ICU Converter
-----
001 ISO-8859  Latin1      ISO 8859-1 Western European      20    057
ISO-8859-1
002 IBM-1047  IBM1047     IBM EBCDIC US English IBM1047      40    213
ibm-1047_P100-1995
003 IBM-037   IBM037      IBM EBCDIC US English IBM037        40    189
ibm-37_P100-1995
....
```

The following table describes the columns in this report:

Column	Description
CPN	PowerExchange code page number.
Name	Name of the code page. PowerExchange and PowerCenter exchange data using this code page name.
PMlocale	Name of the character set.
PowerCenter name	Code page description.
Space	Hexadecimal value of the space character used to pad CHAR columns.
CPN2	PowerExchange code page number of the equivalent ICU code page.
ICU Converter name	Converter name of the equivalent ICU code page.

Report 2. PowerExchange Control Table for ICU Code Pages

This report displays details of all code pages known to PowerExchange.

The following code page number ranges are reserved for ICU code pages:

- The CPN range of 41 to 269. This code page range is used for the ICU code pages that PowerExchange supports, and that are present in the ICU data library, icudt32.dll.
- The CPN range of 270 to 300.
- The CPN range of 301 to 340.

Creating a Custom ICU Code Page

You can create a custom ICU code page.

To create a custom ICU code page:

1. Define the mappings in an ICU UCM file.
2. Run the makeconv utility to generate a binary CNV file from the UCM file.
3. Enter the name of the directory where the binary CNV file is located in the ICUDATADIR parameter of the DBMOVER configuration file.
4. Use the ICUCONVERTER=(301, ADD...) parameter in the DBMOVER configuration file to add the converter to the PowerExchange code page control table.

PowerExchange Control Table for ICU Code Pages Sample Report

The following sample report shows the PowerExchange control table for ICU code pages.

```
Report 2 : Power Exchange control table for ICU code pages
=====

NUM_STATIC_CODEPAGES      = 41
number of ICU code pages = 300
-----
NUMBER_OF_CODEPAGES      = 341

Min      : Minimum bytes per character
Max      : Maximum bytes per character
In       : Increment to column size on top of standard formula
          column_size = to_max / (float)fr_min + 0.5)
Al       : Number of aliases

L        : Length of the space character
Space    : Hex of the space character used to pad CHAR columns

Flag 1 : S = Are States Used
Flag 2 : U = Endianness Unknown
Flag 3 : A = ASCII, E = EBCDIC
Flag 4 : P = SQL is parsable without translation
Flag 5 : L = Lower case not standard for ASCII/EBCDIC SQL
Flag 6 : X = Supplemental characters beyond UCS_2 plain

CPN Converter name  Min Max In Al L Space  Flags  PMlocale  PowerCenter Name
---
041 UTF-8          1  3  0 10 1 20      A P  X  UTF-8    UTF-8 encoding of Unicode
042 UTF-16         2  2  2  0 2 0020      U  X  UTF-16    UTF-16 encoding of
Unicode
043 UTF-16BE       2  2  0 11 2 0020      X  UTF-16BE  UTF-16 encoding of
Unicode (Big Endian)
....
```

The following table describes the fields in the report:

Field	Description
CPN	PowerExchange code page number.
Converter name	ICU converter name.
Min	Minimum bytes per character.
Max	Maximum bytes per character.
In	Increment added to the buffer size in addition to the standard formula. This increment allows space for leading escape sequences on code pages.
Al	Number of aliases for this code page.
L	Length of the space character.
Space	Hexadecimal value of the space character that is used to pad CHAR columns.
Flag 1 : States	Set to S if state bytes are used. For example: <ul style="list-style-type: none"> - X'0E', X'0F' in EBCDIC mixed-length code pages - Code pages with sub-versions such as ISO2022 and ISCII
Flag 2 : Endianness unknown	Set to U if the code page takes a different meaning depending on the endian format of integers. For example, UTF-16 on Windows means UTF-16LE, but UTF-16 on z/OS means UTF-16BE.
Flag 3 : ASCII / EBDIC	Set to A if characters 0 through 9 and A through Z are compatible with 7-bit ASCII. Set to E if characters 0 through 9 and A through Z are compatible with standard EBCDIC, such as IBM-37.
Flag 4 : Parsable without translation	Set to P if the invariant SQL characters are consistent with the standard for the ASCII / EBCDIC type. This flag is used within the ODBC layer to determine whether SQL is understandable on that system. For example, if PowerCenter passes SQL in IBM EBCDIC Japanese CP939 to a Windows program compiled in code page ISO-8859, the SQL must be translated into an ASCII code page before key words like "select," "from," "where" can be recognized.
Flag 5: Lower case non-standard	Set to L if lowercase a to z characters are not consistent with standard EBCDIC such as IBM-37. This problem affects some Japanese code pages. Sometimes this problem is handled by forcing SQL into uppercase before converting it.
Flag 6: Supplemental Characters	Set to X if the code page includes supplemental characters beyond UCS_2.
PMlocale	Name of the code page as it is understood by PowerCenter.
PowerCenter Name	PowerCenter name of the code page. Data can be passed between PowerExchange and PowerCenter in this code page.

Report 3. Comparing Name and Character Size Information with ICU

This report displays the following information for each code page defined in the PowerExchange code page control table:

- **Type of converter.** This value is obtained from ICU.
- **Canonical name of the converter.** This value is obtained from ICU and is printed if it differs from the PowerExchange code page name.
- **Minimum and maximum bytes for each character.** These values are obtained from ICU. If they differ from the values used by PowerExchange, the difference is flagged.

If the code page is not present in the ICU data library DLL and no custom ICU code page CNV file occurs in the ICU data directory, U_FILE_ACCESS_ERROR is issued in an error message.

The following sample report compares name and character size information with ICU.

```
Report 3 : Comparing PWX name, character size information with ICU
=====

ICU version      = 3.2

If different from control array...
Min              = I18N_min_char_size result
Max              = I18N_max_char_size result
Canonical name   = I18N_canon_conv_name result

CPN Converter name          Convert. type Min Max Supplementals
--- -----
041 UTF-8                   UTF8                      contains supplementals
042 UTF-16                  UTF16                     contains supplementals
043 UTF-16BE                UTF16_BigEnd.           contains supplementals
....
```

The following table describes the fields in the report:

Field	Description
CPN	PowerExchange code page number.
Converter name	ICU converter name.
Convert. type	Type of the ICU converter.
Min	Minimum bytes for each character according to the ICU API, if different from the PowerExchange value.
Max	Maximum bytes for each character according to the ICU API, if different from the PowerExchange value.
Supplementals	Supplemental information.

Report 4. Comparing ICU Code Page Information with PowerExchange

This report displays the number of ICU converters.

ICUCHECK.EXE makes iterative passes through all the code pages in the ICU data library using a method similar to the ICUINFO.EXE program. For each code page, it finds the matching entry in the PowerExchange code page control table and lists its CPN and PowerCenter name.

If no matching entry is found in PowerExchange, the following message is issued:

```
*** ICU converter not used
```

If a matching entry is found, the following report is generated.

```
Report 4 : Comparing ICU code page information with PWX
=====

Number of ICU converters=235

Converter name          Min Max Converter type      CPN
-----
UTF-8                  1   3 UTF8                      041
UTF-16                 2   2 UTF16                     042
UTF-16BE               2   2 UTF16_BigEndian          043
....
```

The following table describes the columns in the report:

Column	Description
Converter name	ICU converter name.
Min	Minimum bytes for each character according to the ICU API, if different from the PowerExchange value.
Max	Maximum bytes for each character according to the ICU API, if different from the PowerExchange value.
Converter type	Type of ICU converter.
CPN	PowerExchange code page number.

Report 5. PowerExchange Code Page Names and Aliases

This report displays the PowerExchange number associated with a code page. The report is arranged in ascending order by alias name.

PowerExchange uses aliases in the following situations:

- For DB2, PowerExchange adds the prefix "cp" in front of CCSID numbers to form aliases that are used to find the code pages. For example, a DB2 CCSID of 300 is changed to an alias of "cp300" before PowerExchange looks up the aliases to find PowerExchange CPN 252.
- You can specify the PowerExchange CODEPAGE configuration parameter with aliases, such as:

```
CODEPAGE=(IBM-37,CP930,CP930)
```

When using aliases in these situations, the "cp" format of alias or the PowerCenter name is recommended.

EBCDIC code pages are less common than ASCII. EBCDIC code pages usually have the literal "EBCDIC" in the alias. The following example shows aliases of Denmark EBCDIC Denmark code pages:

```
191 EBCDIC-CP-DK
191 ebcdic-dk
225 ebcdic-dk-277+euro
```

Do not define the same alias on multiple code pages. Only the code page with the lower CPN number is used.

You can use the ICUALIAS parameter to change the alias definitions.

The following sample report shows PowerExchange code page names and aliases.

```
Report 5 : Power Exchange Code page Names and Aliases
=====

Number of Aliases = 1142

CPN Alias name                Alias type
---
058 7-bit ASCII              PM CP long description
171 Adobe Latin1 Encoding    PM CP long description
171 AdobeLatin1Encoding
170 AdobeStandardEncoding
....
```

The following table describes the columns in the report:

Title	Description
CPN	PowerExchange code page number.
Alias name	ICU converter name.
Alias type	Type of code page alias.

Report 6. PM Locale Code Pages

The following sample report shows PM locale code pages.

```
Report 6 : PM locale Code Pages
=====

id name                description                PWX codepages
--
1 US-ASCII             7-bit ASCII             US-ASCII
4 Latin1               ISO 8859-1 Western European ISO-8859, ISO-8859-1
5 ISO-8859-2           ISO 8859-2 Eastern European ibm-912_P100-1995
6 ISO-8859-3           ISO 8859-3 Southeast European ibm-913_P100-2000
....
```

The following table describes the columns in the report:

Column	Description
id	ID number of the code page.
name	Name of the code page.
description	Description of the code page.
PWX codepages	Equivalent PowerExchange code page.

Report 7. Control Counts

This report displays summary statistics on the number of code pages by type.

The following report shows control counts.

```
Report 7 : Control counts
=====
```

```
CPN 001-030: Simple code pages
```

16

```

CPN 031-040: User-defined Simple code pages          10
CPN 041-300: ICU code pages                          235
CPN 301-340: User-defined ICU code pages             0
-----
CPN 001-340: Total code pages                        261

Simple code pages with PMLocales                      16
ICU   code pages with PMLocales                      212
-----
Total  code pages with PMLocales                     228

Most aliases for a code page                          11
Code page with the most aliases                      43 (UTF-16BE)

PowerCenter PMLocales not mapped to PWX              32
....

```

Report 8. Errors

The following report shows errors and warnings.

```

Report 8 : Errors
=====

PWX PMLocales not in PowerCenter                    :   3
-----
Total errors                                         :   3
-----

Return code set = 71 because errors were met

```

Code Pages Shipped with PowerExchange

To support different languages, the following types of code pages are shipped with PowerExchange:

- PowerExchange static single-byte code pages
- ICU version 3.2.1 code pages
- Additional Japanese ICU code page

ICU is an open source development project of IBM. For more information about ICU, see ICU Web page at <http://icu.sourceforge.net/userguide/icudata.html>.

PowerExchange Static Code Pages

The following table lists the single-byte static code pages that are shipped with PowerExchange:

Code Page	Typical Use
ISO-8859	UNIX and Windows
IBM-037	USA, Canada, Brazil
IBM-273	Germany, Austria
IBM-277	Denmark, Norway

Code Page	Typical Use
IBM-278	Finland, Sweden
IBM-280	Italy
IBM-284	Spain, Latin America
IBM-285	United Kingdom
IBM-297	France
IBM-424	Modern Hebrew
IBM-500	International
IBM-870	Polish
IBM-1047	Latin 1/Open Systems
MS-1250	PC Regional options setting for Central Europe (Polish)

Note: Code pages beginning with "IBM-" are relevant to z/OS and i5/OS systems.

Code Pages That Handle All Supported Unicode Characters

The following table lists ICU code pages that handle all characters supported by Unicode:

Name	Typical Use	Character Length in Bytes
BOCU-1	Email	3+
CESU-8	Compression	3+
IMAP-mailbox-name	Email	3+
SCSU	Compression	3+
UTF-16	-	2
UTF16_OppositeEndian	-	2
UTF16_PlatformEndian	-	2
UTF-16BE	Databases	2
UTF-16LE	Databases	2
UTF-32	-	4
UTF32_OppositeEndian	-	4
UTF32_PlatformEndian	-	4

Name	Typical Use	Character Length in Bytes
UTF-32BE	-	4
UTF-32LE	-	4
UTF-7	Email	3+
UTF-8	Databases	3+

Code pages UTF-16 and UTF-32 include a Byte Order marker (BOM) sequence of characters preceding the data. These code pages might be useful in encoding an entire file, but are seldom used in databases where UTF-16BE and UTF-16LE are commonly used or where characters are defined in 16 bit integers.

Little-ended (LE) code pages are used on Linux and Windows systems, for example, UTF-16LE.

Big-ended (BE) code pages are used on Unix, i5/OS and z/OS, for example, UTF-16BE.

PowerExchange does not support supplemental characters defined above the basic plain, that is <U00FFFF>, and which are formed by 2 surrogate NUM16 integers. If PowerExchange encounters a supplemental character, PowerExchange treats it as invalid and replaces it with the substitution character.

Code Pages Based on ASCII Encoding

The following table lists ICU code pages that use the 7-bit ASCII values for the characters X'00' to X'7F':

Name	Typical Use	Character Length ins
ibm-1089_P100-1995	Arabic	1
ibm-1256_P110-1997	Arabic	1
ibm-5352_P100-1998	Arabic	1
windows-1256-2000	Arabic	1
ibm-1257_P100-1995	Baltic	1
ibm-5353_P100-1998	Baltic	1
ibm-901_P100-1999	Baltic	1
ibm-914_P100-1995	Baltic	1
ibm-921_P100-1995	Baltic	1
ibm-9449_P100-2002	Baltic	1
ibm-1131_P100-1997	Belarusian	1
ibm-1250_P100-1995	Central & East Europe	1
ibm-912_P100-1995	Central & East Europe	1

Name	Typical Use	Character Length ins
ibm-913_P100-2000	Central & East Europe	1
ibm-5346_P100-1998	Central & East Europe	1
gb18030	Chinese	3+
ibm-964_P110-1999	Chinese	3+
HZ	Chinese (simplified)	3+
ibm-1381_P110-1999	Chinese (simplified)	1 to 2
ibm-1383_P110-1999	Chinese (simplified)	1 to 2
ibm-1386_P100-2002	Chinese (simplified)	1 to 2
ibm-5478_P100-1995	Chinese (simplified)	2
windows-936-2000	Chinese (simplified)	1 to 2
ibm-1373_P100-2002	Chinese (traditional)	1 to 2
ibm-1375_P100-2003	Chinese (traditional)	1 to 2
ibm-950_P110-1999	Chinese (traditional)	1 to 2
windows-950-2000	Chinese (traditional)	1 to 2
ibm-1251_P100-1995	Cyrillic	1
ibm-5347_P100-1998	Cyrillic	1
ibm-915_P100-1995	Cyrillic	1
ibm-902_P100-1999	Estonian	1
ibm-922_P100-1999	Estonian	1
ibm-923_P100-1998	Estonian	1
ibm-1098_P100-1995	Farsi	1
ibm-1253_P100-1995	Greek	1
ibm-4909_P100-1999	Greek	1
ibm-5349_P100-1998	Greek	1
ibm-813_P100-1995	Greek	1
ibm-1255_P100-1995	Hebrew	1
ibm-5351_P100-1998	Hebrew	1

Name	Typical Use	Character Length ins
ibm-916_P100-1995	Hebrew	1
ibm-9447_P100-2002	Hebrew	1
ibm-1006_P100-1995	India	1
ibm-33722_P120-1999	Japanese	3+
ibm-33722_P12A-1999	Japanese	3+
ibm-942_P12A-1999	Japanese	1 to 2
ibm-943_P130-1999	Japanese	1 to 2
ibm-943_P15A-2003	Japanese	1 to 2
ibm-954_P101-2000	Japanese	3+
ibm-1363_P110-1997	Korean	1 to 2
ibm-1363_P11B-1998	Korean	1 to 2
ibm-949_P110-1999	Korean	1 to 2
ibm-949_P11A-1999	Korean	1 to 2
ibm-970_P110-1995	Korean	1 to 2
ibm-971_P100-1995	Korean	2
windows-1361-2000	Korean	1 to 2
windows-949-2000	Korean	1 to 2
ibm-1133_P100-1997	Lao	1
iso-8859_10-1998	Swedish	1
ibm-1162_P100-1999	Thai	1
ibm-874_P100-1995	Thai	1
windows-874-2000	Thai	1
ibm-1254_P100-1995	Turkish	1
ibm-5350_P100-1998	Turkish	1
ibm-920_P100-1995	Turkish	1
ibm-1124_P100-1996	Ukrainian	1
ibm-1125_P100-1997	Ukrainian	1

Name	Typical Use	Character Length ins
ibm-1051_P100-1995	US and international	1
ibm-1252_P100-2000	US and international	1
ibm-367_P100-1995	US and international	1
ibm-5348_P100-1997	US and international	1
ISO-8859-1	US and international	1
US-ASCII	US and international	1
ibm-1129_P100-1997	Vietnamese	1
ibm-1258_P100-1997	Vietnamese	1
ibm-5354_P100-1998	Vietnamese	1

The official 7-bit ASCII characters are defined in code pages US-ASCII and ibm-367_P100-1995.

ICU Code Pages Based on EBCDIC Encoding

ICU code pages that are based on EBCDIC encoding meet either of the following conditions:

- They support the characters a-z, A-Z, and 0-9 in the following standard EBCDIC ranges: x'81 to x'a9', x'c1' to x'e9, and x'f0 to x'f9.
- They are double-byte characters that are used with standard EBCDIC code pages.

The following table lists these code pages:

Name	Typical Use	Character Length in Bytes
ibm-16804_X110-1999	Arabic	1
ibm-16804_X110-1999,swaplfnl	Arabic	1
ibm-420_X120-1999	Arabic	1
ibm-918_P100-1995	Arabic	1
ibm-1112_P100-1995	Baltic	1
ibm-500_P100-1995	Belgium, Switzerland	1
ibm-1156_P100-1999	Central & East Europe	1
ibm-13124_P100-1995	Chinese (simplified)	1
ibm-1388_P103-2001	Chinese (simplified)	1 to 2
ibm-4933_P100-2002	Chinese (simplified)	2

Name	Typical Use	Character Length in Bytes
ibm-836_P100-1995	Chinese (simplified)	1
ibm-837_X100-1995	Chinese (simplified)	2
ibm-935_P110-1999	Chinese (simplified)	1 to 2
ibm-1159_P100-1999	Chinese (traditional)	1
ibm-1371_P100-1999	Chinese (traditional)	1 to 2
ibm-835_X100-1995	Chinese (traditional)	2
ibm-937_P110-1999	Chinese (traditional)	1 to 2
ibm-1025_P100-1995	Cyrillic	1
ibm-1123_P100-1995	Cyrillic	1
ibm-1154_P100-1999	Cyrillic	1
ibm-1158_P100-1999	Cyrillic	1
ibm-1142_P100-1997	Danish	1
ibm-1142_P100-1997,swaplfnl	Danish	1
ibm-277_P100-1995	Danish	1
ibm-1137_P100-1999	Devanagari	1
ibm-1122_P100-1999	Estonian	1
ibm-1157_P100-1999	Estonian	1
ibm-1097_P100-1995	Farsi	1
ibm-1147_P100-1997	French	1
ibm-1147_P100-1997,swaplfnl	French	1
ibm-297_P100-1995	French	1
ibm-1141_P100-1997	German	1
ibm-273_P100-1995	German	1
ibm-4971_P100-1999	Greek	1
ibm-875_P100-1995	Greek	1
ibm-12712_P100-1998	Hebrew	1
ibm-12712_P100-1998,swaplfnl	Hebrew	1

Name	Typical Use	Character Length in Bytes
ibm-424_P100-1995	Hebrew	1
ibm-4899_P100-1998	Hebrew	1
ibm-803_P100-1999	Hebrew	1
ibm-1149_P100-1997	Icelandic	1
ibm-1149_P100-1997,swaplfnl	Icelandic	1
ibm-871_P100-1995	Icelandic	1
ibm-1144_P100-1997	Italian	1
ibm-1144_P100-1997,swaplfnl	Italian	1
ibm-280_P100-1995	Italian	1
ibm-1390_P110-2003	Japanese	1 to 2
ibm-1399_P110-2003	Japanese	1 to 2
ibm-16684_P110-2003	Japanese	2
ibm-290_P100-1995	Japanese	1
ibm-5123_P100-1999	Japanese	1
ibm-8482_P100-1999	Japanese	1
ibm-930_P120-1999	Japanese	1 to 2
ibm-939_P120-1999	Japanese	1 to 2
infa-FujitsuJEF	Japanese	1 to 2
infa-FujitsuJEF_kana	Japanese	1 to 2
infa-HitachiKEIS	Japanese	1 to 2
infa-HitachiKEIS_kana	Japanese	1 to 2
infa-jipse	Japanese	1 to 2
infa-jipse_kana	Japanese	1 to 2
infa-jp_ebcdic	Japanese	1 to 2
infa-jp_ebcdik	Japanese	1 to 2
infa-melcom	Japanese	1 to 2
infa-melcom_kana	Japanese	1 to 2

Name	Typical Use	Character Length in Bytes
infa-unisys	Japanese	1 to 2
infa-unisys_kana	Japanese	1 to 2
ibm-13121_P100-1995	Korean	1
ibm-1364_P110-1997	Korean	1 to 2
ibm-833_P100-1995	Korean	1
ibm-834_P100-1995	Korean	2
ibm-933_P110-1995	Korean	1 to 2
ibm-1132_P100-1998	Lao	1
ibm-870_P100-1995	Polish	1
ibm-1153_P100-1999	Romanian	1
ibm-1153_P100-1999,swaplfnl	Romanian	1
ibm-1145_P100-1997	Spanish	1
ibm-1145_P100-1997,swaplfnl	Spanish	1
ibm-284_P100-1995	Spanish	1
ibm-1143_P100-1997	Swedish	1
ibm-1143_P100-1997,swaplfnl	Swedish	1
ibm-278_P100-1995	Swedish	1
ibm-1160_P100-1999	Thai	1
ibm-838_P100-1995	Thai	1
ibm-1026_P100-1995	Turkish	1
ibm-1155_P100-1999	Turkish	1
ibm-1146_P100-1997	UK English	1
ibm-1146_P100-1997,swaplfnl	UK English	1
ibm-285_P100-1995	UK English	1
ebcdic-xml-us	US and international	1
ibm-1047_P100-1995	US and international	1
ibm-1047_P100-1995,swaplfnl	US and international	1

Name	Typical Use	Character Length in Bytes
ibm-1140_P100-1997	US and international	1
ibm-1140_P100-1997,swaplnl	US and international	1
ibm-1148_P100-1997	US and international	1
ibm-1148_P100-1997,swaplnl	US and international	1
ibm-37_P100-1995	US and international	1
ibm-37_P100-1995,swaplnl	US and international	1
ibm-1130_P100-1997	Vietnamese	1
ibm-1164_P100-1999	Vietnamese	1

These code pages are based on IBM definitions and are used on i5/OS and z/OS.

IBM frequently describes code pages using CCSID numbers. PowerExchange uses aliases that begin with CP to match the CCSIDs. For example, "CP930" matches CCSID 930 and the ICU converter ibm-930_P120-1999.

IBM frequently indicates revisions by adding 4096 to the CCSID number, for example:

CCSID 300 and CCSID 16684 are related ($16684 = 300 + 4096 * 4$)

The following table lists CCSIDs that you can use in the same DB2 table on i5/OS or z/OS:

Typical Use	Single-Byte SCCSID Columns	Double-Byte GCCSID Columns	Mixed-Length MCCSID Columns
Japan - Extended English	1027	300	939
Japan - Extended Katakana	290	300	930
Japan - Katakana - Kanji	8482	16684	1390
Japan - Latin - Kanji	5123	16684	1399
Korean	833	834	933
Korean	13121	4390	1364
Simplified Chinese	836	837	935
Simplified Chinese	13124	4933	1388
Traditional Chinese	28709	835	937
Traditional Chinese (IBM Big-5)	1114	947	950

Specialized ICU Code Pages

The following table lists ICU code pages that are used in specialized contexts:

Name	Typical Use	Character Length in Bytes
ibm-1276_P100-1995	Adobe	1
ibm-1277_P100-1995	Adobe	1
macos-0_2-10.2	Apple - US & International	1
macos-2566-10.2	Apple - East Asia	1 to 2
macos-29-10.2	Apple - Central & East Europe	1
macos-35-10.2	Apple -Turkish	1
macos-6-10.2	Apple - Greek	1
macos-7_3-10.2	Apple - Cyrillic	1
ibm-437_P100-1995	DOS	1
ibm-850_P100-1995	DOS	1
ibm-851_P100-1995	DOS -Greek	1
ibm-852_P100-1995	DOS	1
ibm-855_P100-1995	DOS - Cyrillic	1
ibm-856_P100-1995	DOS -Hebrew	1
ibm-857_P100-1995	DOS - Turkish	1
ibm-858_P100-1997	DOS	1
ibm-860_P100-1995	DOS - Portuguese	1
ibm-861_P100-1995	DOS - Icelandic	1
ibm-862_P100-1995	DOS - Hebrew	1
ibm-863_P100-1995	DOS - Canadian French	1
ibm-864_X110-1999	DOS - Arabic	1
ibm-865_P100-1995	DOS - Nordic	1
ibm-866_P100-1995	DOS - Russian	1
ibm-867_P100-1998	DOS - Hebrew	1
ibm-868_P100-1995	DOS	1

Name	Typical Use	Character Length in Bytes
ibm-869_P100-1995	DOS - Greek	1
ibm-878_P100-1996	DOS - Russian	1
ibm-897_P100-1995	DOS - Japanese	1
ISCII,version=0	ISCII - Devanagari	3+
ISCII,version=1	ISCII - Bengali	3+
ISCII,version=2	ISCII - Gurmukhi	3+
ISCII,version=3	ISCII - Gujarati	3+
ISCII,version=4	ISCII - Orayi	3+
ISCII,version=5	ISCII - Tamil	3+
ISCII,version=6	ISCII - Telugu	3+
ISCII,version=7	ISCII - Kanada	3+
ISCII,version=8	ISCII - Malayalam	3+
ISO_2022,locale=ja,version=0	ISO2022 - Japanese	3+
ISO_2022,locale=ja,version=1	ISO2022 - Japanese	3+
ISO_2022,locale=ja,version=2	ISO2022 - Japanese	3+
ISO_2022,locale=ja,version=3	ISO2022 - Japanese	3+
ISO_2022,locale=ja,version=4	ISO2022 - Japanese	3+
ISO_2022,locale=ko,version=0	ISO2022 - Korean	3+
ISO_2022,locale=ko,version=1	ISO2022 - Korean	3+
ISO_2022,locale=zh,version=0	ISO2022 - Chinese	3+
ISO_2022,locale=zh,version=1	ISO2022 - Chinese	3+
LMBCS-1	Lotus Notes	1 to 2
LMBCS-11	Lotus Notes - Thai	1 to 2
LMBCS-16	Lotus Notes - Japanese	1 to 2
LMBCS-17	Lotus Notes - Korean	1 to 2
LMBCS-18	Lotus Notes - Chinese	1 to 2
LMBCS-19	Lotus Notes - Chinese	1 to 2

Name	Typical Use	Character Length in Bytes
LMBCS-2	Lotus Notes - Greek	1 to 2
LMBCS-3	Lotus Notes - Hebrew	1 to 2
LMBCS-4	Lotus Notes - Arabic	1 to 2
LMBCS-5	Lotus Notes - Cyrillic	1 to 2
LMBCS-6	Lotus Notes - Central & East Europe	1 to 2
LMBCS-8	Lotus Notes - Turkish	1 to 2

Handling Conversion Errors and Special Situations

This section describes how to handle conversion errors and use a code page other than the one a database expects. This information is based on real-life situations.

Code Page Conversion Errors

If it is not possible to convert a character from one code page to another, ICU replaces the character with the substitution value for the current code page by default. Typically, this substitution value is a hexadecimal value, such as x'1A' or x'3F'.

The following situations can prevent character conversion:

- The character is not valid for the source code page. This situation can be caused by describing the data in the wrong code page.
- The character is valid in the source code page, but no equivalent character exists in the target code page. This situation is common if converting from multibyte characters to a single-byte target code page.

PowerExchange aborts when connecting to databases or describing tables if a code page conversion error occurs during initialization. PowerExchange uses substitution characters if any column or parameter data does not convert.

The following table summarizes conversion errors by context:

Data Type and Context	Result	Action
Names of databases, tables, columns, and files during initialization.	Processing aborts.	Check the control code page.
User ID and password during connect processing.	Processing aborts.	Check the control code page.
SQL during Describe processing.	Processing aborts.	Check the SQL code page.

Data Type and Context	Result	Action
Column data during select processing.	Unrecognized characters are replaced with the substitution characters of the target code page which are hexadecimal values. Processing continues.	Check the data code page Verify that the source data is described correctly using DTLDescribe columns.
Parameter data during delete, insert, select, and update processing.	Unrecognized characters are replaced with the substitution characters of the target code page which are hexadecimal values. Processing continues.	Check the data code page. Verify that the source and target data are described correctly using DTLDescribe columns.

Unexpected Data in a Code Page

If you load data in a code page other than the one that the database metadata specifies, problems might occur when you try to convert data using PowerExchange or any conversion tool that the database provides. If the code page is already supported, you do not need to add a user-defined code page.

You can use the ICUALIAS statement in the DBMOVER configuration file on the system where the PowerExchange Listener runs to replace the default internal code page number with the required internal code page number. Because this procedure remaps all data for the problem CCSID to the new code page, you might need to run multiple PowerExchange Listeners on different ports if you have problem data in multiple code pages.

You can also associate individual data maps with different code pages. For CDC processing, you can then merge the DB2 data maps into the corresponding extraction maps. Using this method, you do not need to use multiple Listeners on different ports.

To test your code page mappings, you can first run a database row test. If the row test produces the expected results, you can then create a PowerCenter workflow with a flat file target in a UTF8 code page. If that workflow produces the desired results, you can then create a workflow with a relational target.

Example 1. Using ICUALIAS to Remap CCSID 37 to Thai 878

In this example, you need to consolidate data from different locations into DB2 for i5/OS tables. The CCSID is set to 37, but the data is in the EBCDIC Thai code page at internal code page number 222, which uses ICU converter ibm-838_P100-1995.

By adding ICUALIAS statements to the DBMOVER configuration file, we can remap CCSID 37 to the Thai code page for all DB2 tables accessed with the associated PowerExchange Listener.

Use the following steps to remap the data and extract it to a flat file in UTF8:

1. Create a separate PowerExchange Listener for each problem code page. Make sure that each PowerExchange Listener has a different i5/OS port and DBMOVER configuration file.
2. Include the following statements in the DBMOVER configuration file for the PowerExchange Listener that is used to access the Thai data:

```
ICUALIAS=(003,DELETE,CP37)
ICUALIAS=(189,DELETE,CP37)
ICUALIAS=(222,ADD,CP37)
```

These statements, respectively, accomplish the following results:

- Remove the default mapping between CCSID 37 and the Internal code page 3 with non-ICU converter name IBM-037
 - Remove the default mapping between CCSID 37 and the internal code page 189 with ICU converter name ibm-37_P100-1995
 - Add a mapping between CCSID 37 and the internal code page number 222 with ICU converter name ibm-838_P100-1995
3. From the PowerExchange Navigator, run the DTLDESCRIBE COLUMNS command to verify that the data is described using the required internal code page number.
 4. From the PowerExchange Navigator, run a database row test to view the data.
 5. Create and run a PowerCenter workflow to copy the data to a flat file in UTF-8. Then view the flat file to verify that no characters were lost.

Example 2. Using DB2 Data Maps to Remap CCSID 37 to Thai CCSID 87

In this example, as in the previous one, you need to consolidate data from different locations into DB2 for i5/OS tables. The CCSID is set to 37, but the data is in the EBCDIC Thai code page at internal code page number 222 which uses ICU converter ibm-838_P100-1995.

1. Add a data map with an access method of DB2. Select the **Import Record Definitions** option to import the columns for the table.
2. Amend the code pages for each CHAR and VARCHAR column to the required code page, or set the required code page in the data map properties so it affects all CHAR and VARCHAR columns in the map.
3. From the PowerExchange Navigator, run the DTLDESCRIBE COLUMNS command to verify that the data is described using the required internal code page number.
4. From the PowerExchange Navigator, run a database row test to view the data.
5. Create and run a PowerCenter workflow to copy the data to a flat file in UTF-8. Then view the flat file to verify that no characters were lost.
6. If you are capturing DB2 change data, add a user-defined extraction map. To add a user-defined extraction map, open the extraction group in the PowerExchange Navigator. On the Extraction Group tab in the Resource Explorer, right-click the extraction group and click **Add Extract Definition**.

After you add the extraction map, merge the data map with the extraction map. In this way the change data will also use the changed code page.
7. Repeat steps 3 through 5 with the DB2 change data.

Type 1 Fallback Mappings and Multibyte Data

For a PowerCenter session that writes data in a non-IBM code page to a z/OS or i5/OS target system, you might achieve more accurate character conversions by enabling type 1 fallback mappings. To enable type 1 fallback mappings, define USE_TYPE1_FALLBACKS=Y in the DBMOVER configuration file.

For example, if a PowerCenter session writes data from a SQL Server data source to a PowerExchange z/OS target, different Unicode values are used for punctuation characters such as the half-width tilde and the half-width dash. If you do not enable type 1 fallback mappings, this situation can result in substitution characters on the z/OS target. However, if you define USE_TYPE1_FALLBACKS=Y in the DBMOVER file on the Linux,

UNIX, or Windows system, the z/OS target receives the closest available match instead of the problem substitution characters.

CHAPTER 11

Using the PowerExchange ODBC Drivers

This chapter includes the following topics:

- [PowerExchange ODBC Driver Overview, 383](#)
- [PowerExchange ODBC Data Sources on Windows, 383](#)
- [PowerExchange ODBC Data Sources on Linux or UNIX, 399](#)
- [SQL Escape Sequences, 414](#)

PowerExchange ODBC Driver Overview

PowerExchange provides thin ODBC drivers that PowerCenter and other clients can use to connect to a data source through PowerExchange.

To define a PowerExchange ODBC data source on Linux or UNIX, edit the `odbc.ini` file.

To define a PowerExchange ODBC data source on Windows, use the **Windows ODBC Data Source Administrator**.

PowerExchange also defines SQL escape sequences that you can use to override PowerExchange defaults when you connect to a data source by using a PowerExchange ODBC driver.

PowerExchange ODBC Data Sources on Windows

Use the PowerExchange ODBC drivers to access PowerExchange ODBC data sources.

Before you create PowerExchange ODBC data sources on Windows, install 32-bit and 64-bit versions of the following PowerExchange ODBC drivers:

- Informatica PowerExchange ODBC Driver. This driver accesses data and metadata for PowerExchange sources and targets.
- Informatica PowerExchange Unicode ODBC Driver. This driver accesses metadata for certain PowerExchange multibyte sources and targets.

For more information about installing these drivers, see the *PowerExchange Installation and Upgrade Guide*.

After you install the PowerExchange ODBC drivers on Windows, use the **Windows ODBC Data Source Administrator** to define PowerExchange ODBC data sources. Also, after you upgrade to a new PowerExchange release on Windows, use the **Windows ODBC Data Source Administrator** to update PowerExchange ODBC data sources that you defined in an earlier release.

Create ODBC data sources on the following machines:

- On the PowerCenter Client machine, create 32-bit ODBC data sources so that PowerCenter can import the metadata for PowerExchange sources and targets.
- If you are using Informatica Data Archive, create 32-bit ODBC data sources on the Data Archive Server.
- On the machine that runs the PowerCenter Integration Service, create 64-bit ODBC data sources so that PowerCenter workflows can connect to the PowerExchange Listener and extract data for PowerExchange sources or targets.

Creating an ODBC Data Source on Windows

Use the **ODBC Data Source Administrator** on a 32-bit or 64-bit system to create a 32-bit or 64-bit PowerExchange ODBC data source on Windows to be used with the Informatica PowerExchange ODBC Driver.

1. To start the **ODBC Data Source Administrator**, perform one of the following actions:
 - If you plan to create a 32-bit data source on a Windows 32-bit system or a 64-bit data source on a Windows 64-bit system, open the **Control Panel** and click **Administrative Tools**. Then double-click **Data Sources (ODBC)**.
 - If you plan to create a 32-bit data source on a Windows 64-bit system, enter the following command at the command prompt:

```
%windir%\SysWOW64\odbcad32.exe
```

Enter this command if you installed the PowerCenter Client, which is a 32-bit application, on a Windows 64-bit system.

The **ODBC Data Source Administrator** dialog box appears.

2. On the **System DSN** tab, click **Add**.
3. In the **Create New Data Source** wizard, select the Informatica PowerExchange driver from the list of available drivers and click **Finish**.

If this driver is not available, see the *PowerExchange Installation and Upgrade Guide*.

The **PowerExchange Data Source** dialog box appears.

4. On the **PowerExchange Data Source** tab, define the ODBC data source.
Enter values for all of the properties that are displayed for the selected data source type.
5. On the **General** tab, enter optional information that does not depend on the data source type.
6. On the appropriate **Properties** tab, enter properties that are specific to the data source type or access method.

The following table identifies the tabs on which to enter the information by data source or access type:

Data Source or Access Type	Properties Tab
CAPX	CAPX Properties tab
CAPXRT	CAPXRT Properties tab
DB2 for i5/OS	AS/400 Properties tab
DB2 for z/OS	DB2/S390 Bulk Load Properties tab

- Click **OK**.

The ODBC data source appears in the **System Data Sources** list on the **System DSN** tab.

- Click **OK**.

RELATED TOPICS:

- [“General Tab” on page 389](#)
- [“AS/400 Properties Tab” on page 392](#)
- [“CAPX Properties Tab” on page 393](#)
- [“CAPXRT Properties Tab” on page 395](#)
- [“DB2/S390 Bulk Load Properties Tab” on page 397](#)
- [“PowerExchange Data Source Tab” on page 386](#)

Creating an ODBC Data Source on Windows to Be Used with the PowerExchange ODBC Unicode Driver

You can access multibyte metadata with the Informatica PowerExchange Unicode driver. Use the **ODBC Data Source Administrator** to create a PowerExchange ODBC data source on Windows, and specify the Informatica PowerExchange Unicode driver.

- To open the **ODBC Data Source Administrator**, perform one of the following actions:
 - If you plan to create a 32-bit data source on a Windows 32-bit system or a 64-bit data source on a Windows 64-bit system, open the **Control Panel** and click **Administrative Tools**. Then double-click **Data Sources (ODBC)**.
 - If you plan to create a 32-bit data source on a Windows 64-bit system, enter the following command at the command prompt:

```
%windir%\SysWOW64\odbcad32.exe
```

Enter this command if you installed the PowerCenter Client, which is a 32-bit application, on a Windows 64-bit system.

The **ODBC Data Source Administrator** dialog box appears.

- On the **System DSN** tab, click **Add**.
The **Create New Data Source** wizard appears.
- Select the Informatica PowerExchange Unicode driver from the list of available drivers and click **Finish**.
If this driver is not available, see the *PowerExchange Installation and Upgrade Guide*.
The **PowerExchange Data Source** dialog box appears.

4. On the **PowerExchange Data Source** tab, enter the following information:
 - In the **Name** box, enter a name for the data source.
 - In the **Local Codepage** list, select the **Unicode for metadata driver** option.
5. Click **OK**.

The ODBC data source appears in the **System Data Sources** list on the **System DSN** tab.
6. Click **OK**.

RELATED TOPICS:

- [“PowerExchange Data Source Tab” on page 386](#)

Updating PowerExchange ODBC Data Sources on Windows

After you upgrade to a new PowerExchange release on Windows and install the PowerExchange ODBC drivers, use the Windows **ODBC Data Source Administrator** to update PowerExchange ODBC data sources that you defined in an earlier release.

Perform this task for each existing PowerExchange ODBC data source.

1. Open the data source in the **ODBC Data Source Administrator**, and select the **System DSN** tab.
2. Select the data source and click **Configure**.
3. In the **ODBC Driver DSN Setup** dialog box, click **OK**.

PowerExchange updates the Windows Registry entry for the data source.

PowerExchange Data Source Wizard Properties

Based on the data source type, you define different properties for the data source in the **PowerExchange Data Source** wizard.

The following topics describe the tabs in the **PowerExchange Data Source** wizard.

PowerExchange Data Source Tab

On the **PowerExchange Data Source** tab in the **PowerExchange Data Source** wizard, enter the following properties to define a PowerExchange ODBC data source:

Name

A name for the data source.

Corresponding ODBC Parameter: Entry in odbc.ini file

Location

The location of the data source.

The locations that appear in the **Location** list are defined by NODE statements in the PowerExchange dbmover.cfg configuration file on the local machine.

Corresponding ODBC Parameter: LOCATION

Type

The data source type.

Note: If you use the DB2400C access method for a DB2 for i5/OS 7.2 source and specify SECURITY=(2,x) in the DBMOVER configuration file, bulk data movement operations that use ODBC fail with an SQL error. This restriction is related to an IBM change to the DB2 interface that the DB2400C access method uses. Use the DB2 access method or specify a different value for the first positional parameter in the SECURITY statement.

Corresponding ODBC Parameter: DBTYPE

Local Codepage

Code page to be used.

Default is **Default**, which causes the code page value in the PowerExchange dbmover.cfg configuration file or the default code page for the system to be used for the code page.

Database

For the MSSQL or ODBC data source type only, the name of the database.

Corresponding ODBC Parameters: For the MSSQL or ODBC data source type, DBQUAL2.

DB Qual2

For the CAPX or CAPXRT data source type only, the application name.

Corresponding ODBC Parameter: DBQUAL2

SSID or DBName

For the DB2, DB2UDB, or DB2400C data source type only, the DB2 SSID or database name.

Corresponding ODBC Parameter: DBQUAL1

Write Mode

For the DB2, DB2UDB, DB2400C, MSSQL, NRDB, NRDB2, ODBC, or ORACLE data source type only, the write mode.

Select one of the following options:

- **Confirm Write On.** Enables confirm write. Ensures that data sent to the PowerExchange ODBC driver is immediately sent to, rather than buffered by, the PowerExchange Listener. Additionally, a success or no success response is sent, which improves error recovery. However, sending a response can slow data transfer rates.
- **Confirm Write Off.** For DB2, DB2 UDB, and Oracle data sources only. Disables confirm write. Before using this setting, contact Informatica Global Customer Support.
- **Asynchronous With Fault Tolerance.** Enables asynchronous write with fault tolerance. For more information about asynchronous write, see the *PowerExchange Bulk Data Movement Guide*.

Default is **Confirm Write On**.

Corresponding ODBC Parameter: CONFIRMWRITE

Reject File

If you select the **Asynchronous With Fault Tolerance** option in the **Write Mode** list, the location and file name of the reject file that is used during asynchronous write operations.

Maximum length is 384 characters. Enter a value of PWXDISABLE to disable reject file logging.

For more information, see the *PowerExchange Bulk Data Movement Guide*.

Corresponding ODBC Parameter: REJECTFILE

Stop On Errors

If you select the **Asynchronous With Fault Tolerance** option in the **Write Mode** list, the number of non-fatal errors that are allowed in the write phase before processing is terminated.

Valid values are from 0 to 2147483647.

For more information, see the *PowerExchange Bulk Data Movement Guide*.

Corresponding ODBC Parameter: STOPONERRORS

SSID

For the DB2390IMG data source type only, the DB2 SSID.

Corresponding ODBC Parameter: DBQUAL1

Wait Time

For the CAPXRT data source types only, the maximum approximate time, in seconds, to wait for data before returning end-of-file (EOF).

Specify one of the following options:

- **0**. The EOF is returned when the end of the current log is reached. The end of the log is determined at the start of the extraction process, because the actual end point constantly changes.
- **2 to 86399**. The number of seconds to wait for data before returning EOF.
- **86400**. The EOF is never returned. The job waits forever.

Corresponding ODBC Parameter: WAITTIME

DSN

For the MSSQL or ODBC data source type only, the DSN name.

Corresponding ODBC Parameter: DBQUAL1

SQL *Net Name

For the ORACLE data source type only, the SQL *Net name.

Corresponding ODBC Parameter: DBQUAL1

RELATED TOPICS:

- [“Creating an ODBC Data Source on Windows” on page 384](#)
- [“PowerExchange ODBC Driver Data Source Parameters” on page 401](#)
- [“AS/400 Properties Tab” on page 392](#)
- [“CAPX Properties Tab” on page 393](#)
- [“CAPXRT Properties Tab” on page 395](#)
- [“DB2/S390 Bulk Load Properties Tab” on page 397](#)
- [“General Tab” on page 389](#)

General Tab

On the **General** tab in the **PowerExchange Data Source** wizard, enter the following optional properties for a PowerExchange data source definition.

Integration Mask

The integration mask. Enter a comma-delimited list of any of the following options:

- CPOOL. Initializes connection pooling.
- DFN. If you specify DFN, the PowerExchange data source type is returned as follows:

PWX:dbtype

If you do not specify DFN, the following string is returned:

PWX

Informatica recommends that you use the default value.

- ILMBRIDGE. Specify this option if the Informatica Data Archive product accesses data via a JDBC-ODBC bridge connection to the PowerExchange ODBC driver. Specify this option to enable the following options: BINASCHAR, CLOSEDSTMREUSE, NOFKEYS, NOUNDERSCORE, SQLENN4BYTES, and V3COLNAMES.
- OWB. Defines various SQL validations and statement reuse properties. Used by Oracle Warehouse Builder.
- V30. Causes the ODBC SQLColumns() call to return V3 column names. Used by the IBM DB2 Information Integrator (II).
- VERBOSE. Causes extra messages to be generated regarding Integration Mask options.

In addition, you can enable the following Integration Mask options by specifying either the specific option or the ILMBRIDGE option:

- BINASCHAR. Causes the ODBC SQLColAttributes() call to change BIN or VARBIN columns to CHAR or VARCHAR columns and double the lengths.
- CLOSEDSTMREUSE. Causes a statement opened in read or select mode to be retained after an ODBC SQLFreeStmt() call of type SQL_CLOSE. Subset of StmtKeepAlive.
- NOFKEYS. Causes the SQLForeignKeys() ODBC call to return an empty result set without a call to the PowerExchange Listener.
- NOUNDERSCORE. Prevents underscore (_) characters from being replaced with the percent sign (%) in DTLDescribe processing. This option allows PowerExchange to process a single NRDB2 table.
- SQLENN4BYTES. Specifies that the JDBC-ODBC bridge calls SQLGetData with a 4-byte integer buffer to return the length of the target data.
- V3COLNAMES. Causes ODBC V3 names to be returned for all metadata result sets, such as SQLTables(), SQLColumns(), and SQLPrimaryKeys(). Superset of the V30 flag.

Corresponding ODBC Parameter: COMPATIBILITY

Maximum Rows

The maximum rows for retrieval.

Valid values are from 0 to 2147483647.

Default is 0, which indicates unlimited rows.

Corresponding ODBC Parameter: MAXROWS

Pacing Size

The number of rows or kilobytes. Set this property when an application, such as an interactive application, cannot keep pace with the flow of data from the PowerExchange Listener.

For maximum performance, set this property to 0.

Default is the value of the API rows_requested parameter.

Corresponding ODBC Parameter: PACESIZE

As KB

Select this option to specify the pacing size in kilobytes.

Default is **As KB**.

Corresponding ODBC Parameter: INTERPRETASROWS

As Rows

Select this option to specify the pacing size in rows.

Default is **As KB**.

Corresponding ODBC Parameter: INTERPRETASROWS

Compress

Select this option to compress data.

This option is cleared by default.

Corresponding ODBC Parameter: COMPRESS

Encrypt

Select this option to encrypt data.

This option is cleared by default.

Corresponding ODBC Parameter: ENCRYPT, as follows:

- If this option is selected, ENCRYPT=N.
- If this option is cleared, ENCRYPT=Y.

Option

The encryption option, which has the following value:

- **AES**. Data is encrypted by using the Advanced Encryption Standard (AES) algorithm.

Note: The values of **DES** and **RC2** are deprecated. PowerExchange converts a value of **DES** or **RC2** to **AES**.

If you select an encryption option, you must select an encryption level in the **Level** list.

Corresponding ODBC Parameter: ENCRYPT

Level

Required if you select the **AES** encryption option.

The encryption level.

The options are:

- **1**. Use a 128-bit encryption key.

- **2.** Use a 192-bit encryption key.
- **3.** Use a 256-bit encryption key.

Default is **1** if the **AES** encryption option is selected.

Corresponding ODBC Parameter: ENCRYPTLEVEL

MQ Get Opt

The type of read operation to be performed from an MQSeries queue.

The options are:

- **BROWSE.** Reads a copy of the data on the queue.
- **READ.** Reads the data and removes it from the queue.

Default is **READ**.

Corresponding ODBC Parameter: MQGETOPT

Application

The application name that identifies extracts. Specifying the application name increases the ability to identify individual use of a single connection. This is then available to task control.

Corresponding ODBC Parameter: DTLAPP

PWX Overrides

Optional. The following overrides apply to PowerExchange ODBC connections for bulk data movement or CDC sessions:

```
ARRAYSIZE=array_size
LOWVALUES=N
TCPIP_OP_TIMEOUT=network_operation_timeout
TCPIP_CON_TIMEOUT=connection_timeout
TCPIP_HB_INTERVAL=nnnnn
USE_CATALOG_METADATA=Y
```

If you specify multiple overrides, use a comma (,) as a separator, for example:

```
TCPIP_OP_TIMEOUT=nnn,TCPIP_CON_TIMEOUT=nnn
```

ARRAYSIZE. The DB2 fetch array size, in number of rows, for DB2 bulk data movement operations that use the DB2 access method. The array size pertains to the DB2 multiple-row FETCH statements that PowerExchange uses to read data from DB2 source tables. Valid values are from 1 through 5000. Default is 25.

Note: PowerExchange dynamically lowers the array size when all the following conditions are true:

- The database type is DB2.
- The table contains LOB columns.
- The ARRAYSIZE value is greater than 1.
- Row size * ARRAYSIZE is greater than 16000000 bytes.

LOWVALUES. If you previously set the LOWVALUES statement in the DBMOVER configuration file to Y for PowerExchange Client for PowerCenter (PWXPC) sessions, you can use this ODBC parameter to specify an override of LOWVALUES=N for sessions that use PowerExchange ODBC connections. You must use PWXPC instead of ODBC to preserve hexadecimal '0' values, called *low values*, in source character fields when passing these values to a PowerCenter session for delivery to a VSAM target on z/OS or a sequential file target on Linux, UNIX, Windows, or z/OS.

TCPIP_OP_TIMEOUT specifies a timeout, in seconds, for a network operation. When the client-side application thread detects that the network operation exceeds this timeout interval, PowerExchange ends the connection and issues a timeout error message.

TCPIP_CON_TIMEOUT specifies a timeout, in seconds, for a connection attempt. If PowerExchange cannot establish an ODBC connection within this time interval, PowerExchange issues an error message.

TCPIP_HB_INTERVAL specifies a heartbeat interval, in seconds, that overrides the default TCP/IP heartbeat interval of 507 seconds. If PowerExchange does not receive data before the TCP/IP heartbeat interval elapses, PowerExchange ends the ODBC connection and associated workflow with a heartbeat timeout error. Use this override to prevent dropped connections from TCP/IP heartbeat timeout errors.

USE_CATALOG_METADATA. Specify USE_CATALOG_METADATA=Y to read metadata from the catalog during a DB2 bulk load session, rather than issuing a SELECT statement. Then, when PowerExchange reads the metadata from the catalog, it is not necessary to grant PowerExchange SELECT privileges on the table.

Corresponding ODBC Parameter: PWXOVERRIDES

RELATED TOPICS:

- [“PowerExchange ODBC Driver Data Source Parameters” on page 401](#)
- [“AS/400 Properties Tab” on page 392](#)
- [“CAPX Properties Tab” on page 393](#)
- [“CAPXRT Properties Tab” on page 395](#)
- [“DB2/S390 Bulk Load Properties Tab” on page 397](#)
- [“PowerExchange Data Source Tab” on page 386](#)

AS/400 Properties Tab

On the **AS/400 Properties** tab in the **PowerExchange Data Source** wizard, enter properties to define a PowerExchange DB2 for i5/OS data source.

Isolation Level

The commit scope of the transaction.

The options are:

- **ALL**
- **CHG**
- **CS**
- **NONE**
- **RR**

This option is typically set to **CS**.

Default is **NONE**.

Corresponding ODBC Parameter: ISOLATION

Library List (Unqualified Table)

Space-delimited list of libraries that PowerExchange searches to qualify the first table name, if the table name is unqualified, on a select, insert, delete, or update statement. You can specify a maximum of 34 libraries. Each library name is a maximum length of ten characters.

Note: If you specify a library list in both the **Library List (Unqualified Table)** and **Table/File Override** boxes, and a table is found in both lists, the table found in the **Table/File Override** list takes precedence.

Corresponding ODBC Parameter: LIBRARYLIST

Table/File Override

Space-delimited list of file overrides in the following format:

filename/newlibrary/newfilename/newmembername

Alternatively, you can use the following format:

filename/newlibrary/newfilename/

In this format, the member defaults to *FIRST.

You can specify a maximum of eight files. Each file specification is a maximum length of 43 characters, as follows:

- Ten characters for the *filename* value
- Ten characters for the *newlibrary* value
- Ten characters for the *newfilename* value
- Ten characters for the *newmember* value, if specified
- Three characters for the slash (/) separators

When you use this override, any occurrence of the specified file name in a SQL statement is overridden with the *newlibrary/newfilename/newmembername* combination, regardless of whether the file is qualified or not. This includes any files that are specified in the **Library List (Unqualified Table)** override.

Note: If you specify a library list in both the **Library List (Unqualified Table)** and **Table/File Override** boxes, and a table is found in both lists, the table found in the **Table/File Override** list takes precedence.

Corresponding ODBC Parameter: OVRDBF

RELATED TOPICS:

- [“PowerExchange ODBC Driver Data Source Parameters” on page 401](#)
- [“CAPX Properties Tab” on page 393](#)
- [“CAPXRT Properties Tab” on page 395](#)
- [“DB2/S390 Bulk Load Properties Tab” on page 397](#)
- [“General Tab” on page 389](#)
- [“PowerExchange Data Source Tab” on page 386](#)

CAPX Properties Tab

On the **CAPX Properties** tab in the **PowerExchange Data Source** wizard, enter properties to define a PowerExchange CAPX data source.

Extract Type

The extraction type. The options are:

- **SL.** Start the extraction from the point where the last extraction ended, either by restart tokens for z/OS, or by checkpoint timestamps for i5/OS.
- **RS.** Restart the extraction from the start point of the last extraction. This option enables you to run the same extraction multiple times.

Corresponding ODBC Parameter: CAPXTYPE

Image Type

The capture image type. The options are:

- **BA.** Captures the before image data, before the latest change, and the after image data, after the latest change. The before image data is shipped with an action character forcing the deletion of the record. The after image data, which includes the change data, is in the format of an insert operation.
- **AI.** Captures the after image data only.
- **TU.** Captures the before and after image data, as it occurs. The before image data enables the application to ensure that no changes occurred in the record since the request. The after image data appears as an update operation. The update is a full image, but the application that captures this change can change the target record rather than delete it and insert a new record, as occurs with a BA.

Corresponding ODBC Parameter: CAPXIMAGETYPE

AS400 Library/Filename

Overrides the fully qualified DB2 for i5/OS library and table name specified in the PowerExchange capture registration.

For example:

STQA/NEWTABLE

Corresponding ODBC Parameter: DBQUAL1

AS400 Instance

Overrides the name of the DB2 for i5/OS instance specified in the PowerExchange capture registration.

Corresponding ODBC Parameter: DBQUAL2

CAPI Connection

Overrides the default database connection specified in CAPI_CONNECTION statement in the DBMOVER configuration file.

To point to the relevant data source, specify the value in the NAME parameter of the CAPI_CONNECTION statement in the DBMOVER configuration file.

Corresponding ODBC Parameter: DTLCONN_OVR

Extraction Schema

Overrides the schema name specified in an extraction map.

Maximum length is 128 characters. No spaces are allowed.

Corresponding ODBC Parameter: XTRASchema

RELATED TOPICS:

- [“PowerExchange ODBC Driver Data Source Parameters” on page 401](#)
- [“AS/400 Properties Tab” on page 392](#)
- [“CAPXRT Properties Tab” on page 395](#)
- [“DB2/S390 Bulk Load Properties Tab” on page 397](#)
- [“General Tab” on page 389](#)
- [“PowerExchange Data Source Tab” on page 386](#)

CAPXRT Properties Tab

On the **CAPXRT Properties** tab in the **PowerExchange Data Source** wizard, enter properties to define a PowerExchange CAPXRT data source.

Extract Type

The extraction type. The options are:

- **SL**. Extracts all data since the last extraction.
- **RS**. Restarts the previous, or a specified, extraction.

Corresponding ODBC Parameter: CAPXTYPE

Image Type

The image type. The options are:

- **BA**. Captures the before image data, before the latest change, and the after image data, after the latest change. The before image data is shipped with an action character forcing the deletion of the record. The after image data, which includes the change data, is in the format of an insert operation.
- **AI**. Captures the after image data only.
- **TU**. Captures the before and after image data, as it occurs. The before image data enables the application to ensure that no changes occurred in the record since the request. The after image data appears as an update operation. The update is a full image, but the application that captures this change can change the target record rather than delete it and insert a new record, as occurs with a BA.

Corresponding ODBC Parameter: CAPXIMAGETYPE

AS400 Library/Filename

Overrides the fully qualified DB2 for i5/OS library and table name specified in the PowerExchange capture registration.

For example:

```
STQA/NEWTABLE
```

Corresponding ODBC Parameter: DBQUAL1

AS400 Library/Journal

Overrides the fully qualified library and journal name specified in the PowerExchange capture registration.

For example:

```
STQA/NEWJOURNAL
```

Corresponding ODBC Parameter: UAP

Oracle Instance

Overrides the Oracle instance value in the second positional parameter in the ORACLEID statement in the dbmover.cfg configuration file. For example:

```
ORACLEID=(coll_id,oracle_sid,connect_string,cap_connect_string)
```

Used in conjunction with the **Oracle Connection String** value to enable the use of a single set of capture registrations to capture data from multiple Oracle instances.

If you specify an **Oracle Instance** value, but no **Oracle Connection String** value, Oracle capture uses the Oracle connection string value specified in the ORACLEID statement in the dbmover.cfg file.

Corresponding ODBC Parameter: ORAINST

Oracle Connection String

Overrides the Oracle connection string value in the third positional parameter in the ORACLEID statement in the dbmover.cfg configuration file. For example:

```
ORACLEID=(coll_id,oracle_sid,connect_string,cap_connect_string)
```

Used in conjunction with the **Oracle Instance** value to enable the use of a single set of capture registrations to capture data from multiple Oracle instances.

If you specify an **Oracle Connection String** value, but no **Oracle Instance** value, Oracle capture uses the Oracle connection string value specified in the ORACLEID statement in the dbmover.cfg file.

Corresponding ODBC Parameter: ORACONN

Oracle Schema

Overrides the Oracle schema value in a PowerExchange registration group to enable the use of a single set of capture registrations to capture data from multiple schemas that might exist in an Oracle instance.

Corresponding ODBC Parameter: ORASchema

DB2 UDB Database

For DB2 for Linux, UNIX, and Windows only. Overrides the database connection value specified in the PowerExchange extraction map.

Corresponding ODBC Parameter: DATABASE

CAPI Connection

Overrides the default database connection specified in a CAPI_CONNECTION statement in the DBMOVER configuration file.

To point to the relevant data source, use the value that is specified in the NAME parameter of the CAPI_CONNECTION statement in the DBMOVER configuration file.

Corresponding ODBC Parameter: DTLCONN_OVR

Extraction Schema

Overrides the schema name specified in a PowerExchange extraction map.

Maximum length is 128 characters. No spaces allowed.

Corresponding ODBC Parameter: XTRASchema

RELATED TOPICS:

- [“PowerExchange ODBC Driver Data Source Parameters” on page 401](#)
- [“AS/400 Properties Tab” on page 392](#)
- [“CAPX Properties Tab” on page 393](#)
- [“DB2/S390 Bulk Load Properties Tab” on page 397](#)
- [“General Tab” on page 389](#)
- [“PowerExchange Data Source Tab” on page 386](#)

DB2/S390 Bulk Load Properties Tab

For DB2 data sources only. On the **DB2/S390 Bulk Load Properties** tab in the **PowerExchange Data Source** wizard, enter the properties to define PowerExchange for DB2 bulk load processing.

Bulk Load

Select this option to enable input on this page.

Corresponding ODBC Parameter: BULKLOAD

Load Options

The load option.

The options are:

- **RESUME**. Generates a LOAD RESUME statement.
- **REPLACE**. Generates a LOAD REPLACE statement.

Corresponding ODBC Parameter: LOADOPTIONS, as follows:

- RESUME is equivalent to LOADOPTIONS=1.
- REPLACE is equivalent to LOADOPTIONS=4.

Delete Temporary Files

Indicates whether to delete temporary files. The options are:

- **NO**. Temporary files are not deleted.
- **BEFORE**. Temporary files are deleted before running the loader.
- **AFTER SUCCESS ONLY**. For a mode type of **JOB** only. Temporary files are deleted after running the loader, if the return code is 0.
- **AFTER**. For a mode type of **JOB** only. Temporary files are deleted after running the loader.

Corresponding ODBC Parameter: DELETETEMPFILES, as follows:

- NO is equivalent to DELETETEMPFILES=1.
- BEFORE is equivalent to DELETETEMPFILES=2.
- AFTER SUCCESS ONLY is equivalent to DELETETEMPFILES=3.
- AFTER is equivalent to DELETETEMPFILES=4.

JCL Template

The name of the JCL file template on the target system, which is a member in a PDS defined on the system where the PowerExchange Listener is running.

Default is DB2LDJCL. You must customize this JCL before using it.

Corresponding ODBC Parameter: JCLTEMPLATE

CTL Template

The name of the DB2 bulk load control file template on the target system, which is a member in a PDS defined on the system where the PowerExchange Listener is running.

If you specify a CTL template, the **Load Options** option is ignored.

Default is DB2LDJCL. You must customize this JCL before using it.

Corresponding ODBC Parameter: CTLTEMPLATE

Mode Type

The mode type. The options are:

- **JOB**. Submits as a job to the internal reader.
- **NOSUBMIT**. Not submitted. Instead, creates the DB2 loader-specific control files and data files, saves them on the target system, and generates an executable JCL deck on disk.
- **TASK**. Submits as a subtask of the PowerExchange Listener.

Corresponding ODBC Parameter: MODETYPE as follows:

- TASK is equivalent to MODETYPE=1.
- NOSUBMIT is equivalent to MODETYPE=2.
- JOB is equivalent to MODETYPE=3.

Mode Time

The mode time. The options are:

- **WAIT**. Synchronous. For a mode type of **JOB** or **TASK** only. Report the results at end the end of the job or task.
- **NOWAIT**. Asynchronous. For a mode type of **JOB** or **NOSUBMIT** only. Returns OK when a job is submitted.
- **TIMED**. Synchronous. For a mode type of **JOB** only. Wait for the specified number of seconds.
- **DATAONLY**. For all mode types, but typically used with the **NOSUBMIT** mode type. Creates the data DB2 loader file.

Corresponding ODBC Parameter: MODETIME, as follows:

- WAIT is equivalent to MODETIME=1.
- NOWAIT is equivalent to MODETIME=2.
- TIMED is equivalent to MODETIME=3.
- DATAONLY is equivalent to MODETIME=4.

Time

Valid only for a mode type of **JOB** and a mode time of **TIMED**.

A value from 0 to 99999, as follows:

- **0**. No wait.
- **99999**. Wait.
- Any other value. The number of seconds to wait.

Corresponding ODBC Parameter: TIME

Space

The z/OS space allocation. The options are:

- **TRACK**. Tracks.
- **CYLINDERS**. Cylinders.

Corresponding ODBC Parameter: SPACE

Primary Space

Amount of primary space for z/OS files.

Corresponding ODBC Parameter: PRISPACE

Secondary Space

Amount of secondary space for z/OS files.

Corresponding ODBC Parameter: SECSPACE

FName

The file name on the host system. Used as the stem for files created on z/OS.

The files created are:

- FNAME.DTLnnnnn
Data for load.
- FNAME.DTLnnnnn.CTL
Load commands.
- FNAME.DTLnnnnn.SQL
SQL template - create table.
- FNAME.DTLnnnnn.SYSPRINT
Output from loader.

Corresponding ODBC Parameter: FNAME

RELATED TOPICS:

- [“PowerExchange ODBC Driver Data Source Parameters” on page 401](#)
- [“AS/400 Properties Tab” on page 392](#)
- [“CAPX Properties Tab” on page 393](#)
- [“CAPXRT Properties Tab” on page 395](#)
- [“General Tab” on page 389](#)
- [“PowerExchange Data Source Tab” on page 386](#)

PowerExchange ODBC Data Sources on Linux or UNIX

To create an ODBC data source on Linux or UNIX, update the appropriate `odbc.ini` file.

Note: To specify the location of the `odbc.ini` file, set the `ODBCINI` environment variable to the full path and file name.

RELATED TOPICS:

- [“General PowerExchange ODBC Driver Parameters” on page 402](#)
- [“PowerExchange ODBC Driver for DB2 for z/OS Bulk Parameters” on page 413](#)

Example PowerExchange Data Source Specifications

The PowerExchange ODBC driver data sources are defined in the ODBC Data Sources section and then the PowerExchange ODBC driver parameters are set in the individually named section.

For example the `infadb2` data source parameters are defined in the `[infadb2]` section:

```
[ODBC Data Sources]
infadb2=INFA PWX850 DB2
infanrdb=INFA PWX850 NRDB
loaddb2=INFA PWX850 DB2LOAD
oracleunicode=INFA PWX850 UNICODE

[infadb2]
DRIVER=/usr/powerexchange/bin/libdtlodbcs.l
DESCRIPTION='Descriptive Text for DB2 Data Source section'
DBTYPE=db2
LOCATION=db2
DBQUAL1=dsn1

[infanrdb]
DRIVER=/usr/powerexchange/bin/libdtlodbcs.so
DESCRIPTION='Freeform text for nrdb section'
DBTYPE=nrdb
LOCATION=laptop

[infanrdb2]
DRIVER=/usr/powerexchange/bin/libdtlodbcs.so
DESCRIPTION='Freeform text for nrdb2 section'
DBTYPE=nrdb2
LOCATION=node1

[loaddb2]
DRIVER=/usr/lw/stprod310/libdtlodbcs.so
DESCRIPTION=LoadDB2
DBTYPE=DB2
LOCATION=RemoteLOC                      (From DBMOVER.CFG)
DBQUAL1=SSID                          (target database SSID)
BULKLOAD=Y
CTLTEMPLATE=DB2LDCTL
JCLTEMPLATE=DB2LDJCL
DELETETEMPFILES=2                      (BEFORE)
FNAME=INFA.V310.ODBC
LOADOPTIONS=4                          (REPLACE)
MODETIME=1                             (WAIT)
MODETYPE=3                             (JOB)
PRISPACE=1
SECSpace=1
SPACE=t
TIME=0

[oracleunicode]
DRIVER=/dtlqa2/v850/libdtlodbcsu.so
DESCRIPTION=
DATABASE=
DBTYPE=ORACLE
LOCATION=tortilla
DBQUAL1=TO101UTF
MAXROWS=0
COMPRESS=N
ENCRYPT=N
CONFIRMWRITE=N
PAGESIZE=0
INTERPRETASROWS=N
BULKLOAD=N
DELETETEMPFILES=0
LOADOPTIONS=0
LOCALCODEPAGE=41
MODETYPE=0
```



```
MODETIME=0  
TIME=0
```

Note: If no ODBC Driver Manager is installed, you can use the one provided with the PowerExchange software. This code is provided as libdtlodbinst.so or libdtlodbinst.sl (HP only) so as not to impact any existing ODBC Driver Manager. To use this code, you must either:

- Rename it to libodbcinst.so or libodbcinst.sl (HP only).
- Create a hard or symbolic link to it as libodbcinst.so or libodbcinst.sl (HP only).

Accessing Multibyte Metadata with ODBC

You must use the Informatica PowerExchange Unicode driver and local code page 41 to access multibyte metadata.

In the `odbc.ini` file example shown in [“Example PowerExchange Data Source Specifications” on page 400](#), the `oracleunicode` data source is defined and the following parameters set:

```
DRIVER=/dtlqa2/v850/libdtlodbcu.so  
LOCALCODEPAGE=41
```

RELATED TOPICS:

- [“Example PowerExchange Data Source Specifications” on page 400](#)

PowerExchange ODBC Driver Data Source Parameters

Define PowerExchange ODBC driver parameters in the `ODBC.ini` file on Linux or UNIX.

General PowerExchange ODBC Driver Parameters

Enter general PowerExchange ODBC driver parameters in the `odbc.ini` file on Linux or UNIX.

The following table describes these general parameters:

Parameter	Default	Tokens	Usage
CAPXIMAGETYPE	None	BA, AI, or TU	<p>The capture image type for accessing PowerExchange CDC or live data. The options are:</p> <ul style="list-style-type: none">- BA. Captures the before image data, before the latest change, and the after image data, after the latest change. The before image data is shipped with an action character forcing the deletion of the record. The after image data, which includes the change data, is in the format of an insert operation.- AI. Captures the after image data only.- TU. Captures the before and after image data, as it occurs. The before image data enables the application to ensure that no changes occurred in the record since the request. The after image data appears as an update operation. The update is a full image, but the application that captures this change can change the target record rather than delete it and insert a new record, as occurs with a BA.
CAPXTYPE	None	SL or RS	<p>The type of extraction. The options are:</p> <ul style="list-style-type: none">- SL. Start the extraction from the point where the last extraction ended, either by restart tokens for z/OS, or by checkpoint timestamps for i5/OS.- RS. Restart the extraction from the start point of the last extraction. This option enables you to run the same extraction multiple times.

Parameter	Default	Tokens	Usage
COMPATIBILITY	None	-	<p>The integration mask. Includes the following options:</p> <ul style="list-style-type: none"> - CPOOL. Initializes connection pooling. - DFN. If you specify DFN, the PowerExchange data source type is returned as follows: <i>PWX:dbtype</i> <p>If you do not specify DFN, the following string is returned: <i>PWX</i></p> <p>Informatica recommends that you use the default value.</p> <ul style="list-style-type: none"> - ILMBRIDGE. Specify this option if the Informatica Data Archive product accesses data by means of a JDBC-ODBC bridge connection to the PowerExchange ODBC driver. The ILMBRIDGE option enables the following options: BINASCHAR, CLOSEDSTMREUSE, NOFKEYS, NOUNDERSCORE, SQLEN4BYTES, and V3COLNAMES. - OWB. Defines various SQL validations and statement reuse properties. Used by Oracle Warehouse Builder. - V30. Causes the ODBC SQLColumns() call to return V3 column names. Used by the IBM DB2 Information Integrator (II). - VERBOSE. Causes extra messages to be generated regarding Compatibility options. <p>In addition, you can enable the following COMPATIBILITY options by specifying either the specific option or the ILMBRIDGE option:</p> <ul style="list-style-type: none"> - BINASCHAR. Causes the ODBC SQLColAttributes() call to change BIN or VARBIN columns to CHAR or VARCHAR columns and double the lengths. - CLOSEDSTMREUSE. Causes a statement opened in read or select mode to be retained after an ODBC SQLFreeStmt() call of type SQL_CLOSE. Subset of StmtKeepAlive. - NOFKEYS. Causes the SQLForeignKeys() ODBC call to return an empty result set without a call to the PowerExchange Listener. - NOUNDERSCORE. Prevents underscore (_) characters from being replaced with the percent sign (%) in DTLDescribe processing. This option allows PowerExchange to process a single NRDB2 table. - SQLEN4BYTES. Specifies that the JDBC-ODBC bridge calls SQLGetData with a 4-byte integer buffer to return the length of the target data. - V3COLNAMES. Causes ODBC V3 names to be returned for all metadata result sets, such as SQLTables(), SQLColumns(), and SQLPrimaryKeys(). Superset of the V30 flag.

Parameter	Default	Tokens	Usage
COMPRESS	N	Y, N	Set this parameter to compress data. Default is N.
CONFIRMWRITE	Y	Y, N, or T	For DB2, DB2UDB, DB2400C, MSSQL, NRDB, NRDB2, ODBC, or ORACLE data source types only. The write mode. The options are: <ul style="list-style-type: none"> - Y. Enables confirm write. Ensures that data sent to the PowerExchange ODBC driver is immediately sent to, rather than buffered by, the PowerExchange Listener. Additionally, a success or no success response is sent, which improves error recovery. However, sending a response can slow data transfer rates. - T. Enables asynchronous write with fault tolerance. For more information about asynchronous write, see the <i>PowerExchange Bulk Data Movement Guide</i>. - N. For DB2, DB2 UDB, and Oracle data sources only. Disables confirm write. Before using this setting, contact Informatica Global Customer Support. Default is Y.
DATABASE	-	-	For DB2 for Linux, UNIX, and Windows only. Overrides the database connection value specified in the PowerExchange extraction map.

Parameter	Default	Tokens	Usage
DB2DEGREE	None	Any string of text up to 30 characters	<p>For DB2 for i5/OS and DB2 for z/OS only. Sends the following SQL command to DB2, specifying whether DB2 should use, or not use, inter-partition parallel processing:</p> <pre>SET CURRENT DEGREE</pre> <p>For more information about the effect of each value in the SQL command, see the relevant IBM DB2 documentation for the "SET CURRENT DEGREE" and "CHGQRYA DEGREE" commands.</p> <p>For z/OS only:</p> <p>Any text following the equals sign is taken as text to follow a 'SET CURRENT DEGREE ='.</p> <p>For example, specifying DB2DEGREE=ANY runs the statement "SET CURRENT DEGREE = 'ANY'", and would set DB2 to be able to use parallel processing.</p> <p>For i5/OS only:</p> <p>Any text following the equals sign is taken as text to follow a 'CHGQRYA DEGREE' statement.</p> <p>For example, specifying DB2DEGREE=SYSTEM runs the statement "CHGQRYA DEGREE (*SYSTEM)".</p> <p>To maintain compatibility with previous releases, if the parameter begins with a digit, it is appended to a "CHGQRYA DEGREE (*NBRTASKS)" statement. Thus a "CHGQRYA DEGREE (*NBRTASKS 1)" statement can be generated with either a DB2DEGREE=1 or DB2DEGREE="*NBRTASKS 1" statement.</p>
DBQUAL1	None	See subsequent table.	See subsequent table.
DBQUAL2	None	See subsequent table.	See subsequent table.
DESCRIPTION	None	Text	Descriptive text.
DRIVER	None	Path/File	The path and file name of the PowerExchange ODBC driver. The file name can have the suffix .so or .sl according to the version of UNIX being run. Review the suffixes on the existing PowerExchange shared libraries to choose the correct one.
DTLAPP	-	-	Application name. Specifying the application name that identifies extracts increases the ability to identify individual use of a single connection. This is then available to task control.
DTLCONN_OVR	None	Not applicable	The name in the CAP_CONNECTION statement in the DBMOVER configuration file that points to the relevant data source.

Parameter	Default	Tokens	Usage
DTLIMGOV	N	Y or N	<p>For DB2390IMG only.</p> <p>If the image copy data set is specified, PowerExchange checks that the image copy file exists and that a consistent image copy is selected. If either of these is not true, such as when the image copy file does not exist or is inconsistent, the request is rejected and a message is displayed.</p> <p>You can override this behavior by setting the DTLIMGOV parameter to Y. This setting enables inconsistent image copies to be read.</p>
ENCRYPT	N	N, Y, AES	<p>The encryption option. The options are:</p> <ul style="list-style-type: none"> - N. No encryption is used when moving data. - Y. Data is encrypted by using a special Informatica developed algorithm if it is being accessed by way of a PowerExchange Listener. - AES. Data is encrypted by using Advanced Encryption Standard (DES) algorithm. <p>If you specify an encryption option, you must also specify an encryption level in the ENCRYPTLEVEL parameter.</p> <p>Note: The values of DES and RC2 are deprecated. PowerExchange converts a value of DES or RC2 to AES.</p>
ENCRYPTLEVEL	-	1, 2 or 3	<p>The encryption level. Required if the ENCRYPT parameter is set to DES or RC2.</p> <p>The options are:</p> <ul style="list-style-type: none"> - 1. Use a 128-bit encryption key. - 2. Use a 192-bit encryption key. - 3. Use a 256-bit encryption key.
INTERPRETASROWS	N	Y, N	<p>Indicates whether to interpret the pacing size in rows or kilobytes. The options are:</p> <ul style="list-style-type: none"> - Y. Interpret the pacing size in rows. - N. Interpret the pacing size in kilobytes. <p>Default is Y.</p>
ISOLATION	NONE	ALL, CHG, CS, NONE, or RR	For DB2 for i5/OS only. The commit scope of the transaction. This parameter is typically set to CS.
JRNL	-	LIBRARY/ JOURNAL	<p>For i5/OS only. The fully qualified library and journal name that is to be used instead of that specified in the PowerExchange capture registration.</p> <p>For example:</p> <p>STQA/NEWJOURNAL</p>

Parameter	Default	Tokens	Usage
LIBRARYLIST		Maximum of 34 libraries. Each library name is a maximum length of ten characters.	A space-delimited list of libraries that PowerExchange searches to qualify the first table name, if the table name is unqualified, on a select, insert, delete, or update statement. Note: If you specify a library list in both the LIBRARYLIST and OVRDBF parameters, and a table is found in both lists, the table found in the OVRDBF parameter takes precedence.
LOCALCODEPAGE	None	-	A number that identifies the code page to be used. For example 41 represents UTF-8. Specify NONE to use the code page value specified in the DBMOVER configuration file at run time. You can use the icucheck.exe command-line utility to generate a lookup list of numbers with the corresponding code page names. You can pipe the results of the utility to a file as follows: icucheck.exe > iculist.txt
LOCATION	None	None	The location of the data source. The location is defined by a NODE statement in the dbmover.cfg configuration file on the local system.
MAXROWS	0	0 to 2147483647	The maximum number of rows for retrieval. Default is 0, which indicates unlimited rows.
MQGETOPT	R	B or R	The type of read operation to be performed from an MQSeries queue. The options are: - B. Browse. Reads a copy of the data on the queue. - R. Read. Reads the data and removes it from the queue. Default is R.
NOUPDATECDP	N	Y or N	Indicates whether database row test output is to be added to the PowerExchange CDC audit trail. The options are: - Y. Omit the database row test output from the PowerExchange CDC audit trail for the application. - N. Include the database row test output in the PowerExchange CDC audit trail for the application.
ORACOLL	-	-	Overrides the value specified in the ORACOLL parameter in the ORAD CAPI_CONNECTION statement in the DBMOVER configuration file. Enables the use of a single set of capture registrations to capture data from up to ten Oracle instances at once.

Parameter	Default	Tokens	Usage
ORACONN	-	-	<p>Overrides the Oracle connection string value in the third positional parameter in the ORACLEID statement in the dbmover.cfg configuration file. For example:</p> <pre>ORACLEID=(coll_id, oracle_sid, connect_string, cap_connect_string)</pre> <p>Used in conjunction with the ORAINST value to enable the use of a single set of capture registrations to capture data from multiple Oracle instances.</p> <p>If you specify an ORACONN value without an ORAINST value, Oracle capture uses the Oracle connection string value specified in the ORACLEID statement in the dbmover.cfg file.</p>
ORAINST	-	-	<p>Overrides the Oracle instance value in the second positional parameter in the ORACLEID statement in the dbmover.cfg configuration file.</p> <p>For example:</p> <pre>ORACLEID=(coll_id, oracle_sid, connect_string, cap_connect_string)</pre> <p>Used in conjunction with the ORACONN value to enable the use of a single set of capture registrations to capture data from multiple Oracle instances.</p> <p>If you specify an ORAINST value, but no ORACONN value, Oracle capture uses the Oracle connection string value specified in the ORACLEID statement in the dbmover.cfg file.</p>
ORASchema	-	-	<p>Overrides the Oracle schema value in a PowerExchange registration group to enable the use of a single set of capture registrations to capture data from multiple schemas that might exist in an Oracle instance.</p>

Parameter	Default	Tokens	Usage
OVRDBF	-	<p>Maximum of eight files. Each file specification is a maximum length of 43 characters, as follows:</p> <ul style="list-style-type: none"> - Ten characters for the <i>filename</i> value - Ten characters for the <i>newlibrary</i> value - Ten characters for the <i>newfilename</i> value - Ten characters for the <i>newmember</i> value, if specified - Three characters for the slash (/) separators 	<p>Space-delimited list of file overrides in the following format:</p> <pre>filename/newlibrary/ newfilename/newmembername</pre> <p>Alternatively, you can use the following format:</p> <pre>filename/newlibrary/ newfilename/</pre> <p>In this format, the member defaults to *FIRST.</p> <p>When you use this override, any occurrence of the specified file name in a SQL statement is overridden with the <i>newlibrary/newfilename/newmembername</i> combination, regardless of whether the file is qualified or not. This includes any files that are specified in the LIBRARYLIST override.</p> <p>Note: If you specify a library list in both the LIBRARYLIST and OVRDBF parameters, and a table is found in both lists, the table found in the OVRDBF parameter takes precedence.</p>
PACESIZE	Value of the API rows_request parameter.	Numeric	<p>The number of rows or kilobytes. Set this parameter when an application, such as an interactive application, cannot keep pace with the flow of data from the PowerExchange Listener.</p> <p>For maximum performance, set this parameter to 0.</p>

Parameter	Default	Tokens	Usage
PWXOVERRIDES	None	-	<p>Overrides that apply to PowerExchange ODBC connections for bulk data movement and CDC sessions:</p> <ul style="list-style-type: none"> - ARRAYSIZE. The DB2 fetch array size, in number of rows, for DB2 bulk data movement operations that use the DB2 access method. The array size pertains to the DB2 multiple-row FETCH statements that PowerExchange uses to read data from DB2 source tables. Valid values are from 1 through 5000. Default is 25. <p>Note: PowerExchange dynamically lowers the array size when all the following conditions are true:</p> <ul style="list-style-type: none"> - The database type is DB2. - The table contains LOB columns. - The ARRAYSIZE value is greater than 1. - Row size * ARRAYSIZE is greater than 16000000 bytes. - LOWVALUES. If you previously set the LOWVALUES statement in the DBMOVER configuration file to Y for PowerExchange Client for PowerCenter (PWXPC) sessions, you can use this ODBC parameter to specify an override of LOWVALUES=N for sessions that use PowerExchange ODBC connections. You must use PWXPC instead of ODBC to preserve hexadecimal '0' values, called <i>low values</i>, in source character fields when passing these values to a PowerCenter session for delivery to a VSAM target on z/OS or a sequential file target on Linux, UNIX, Windows, or z/OS. - TCPIP_OP_TIMEOUT. Specifies a timeout, in seconds, for a network operation. When the client-side application thread detects that the network operation exceeds this timeout interval, PowerExchange ends the connection and issues a timeout error message. - TCPIP_CON_TIMEOUT. Specifies a timeout, in seconds, for a connection attempt. If PowerExchange cannot establish an ODBC connection within this time interval, PowerExchange issues an error message. - TCPIP_HB_INTERVAL. Specifies a heartbeat interval, in seconds, that overrides the default TCP/IP heartbeat interval of 507 seconds. If PowerExchange does not receive data before the TCP/IP heartbeat interval elapses, PowerExchange ends the ODBC connection and associated workflow with a heartbeat timeout error. Use this override to prevent dropped connections from TCP/IP heartbeat timeout errors. - USE_CATALOG_METADATA. Specify USE_CATALOG_METADATA=Y to read metadata from the catalog during a DB2 bulk load session, rather than issuing a SELECT statement. Then, when PowerExchange reads the metadata from

Parameter	Default	Tokens	Usage
			<p>the catalog, it is not necessary to grant PowerExchange SELECT privileges on the table.</p> <p>Use the following syntax:</p> <pre>PWXOVERRIDES=ARRAYSIZE=array_size PWXOVERRIDES=LOWVALUES=N PWXOVERRIDES=TCPIP_OP_TIMEOUT=network_operation_timeout PWXOVERRIDES=TCPIP_CON_TIMEOUT=connection_timeout PWXOVERRIDES=TCPIP_HB_INTERVAL=nnnnn PWXOVERRIDES=USE_CATALOG_METADATA=Y</pre> <p>If you specify multiple overrides, use a semicolon (;) as a separator, for example:</p> <pre>PWXOVERRIDES=TCPIP_OP_TIMEOUT=nnn;TCPIP_CON_TIMEOUT=nnn PWXOVERRIDES=LOWVALUES=N</pre>
REJECTFILE	None	Up to 384 characters	<p>Available only if the CONFIRMWRITE parameter is set to T, which enables asynchronous with fault tolerance mode.</p> <p>The location and file name of the reject file that is used during asynchronous write operations.</p> <p>Enter a value of PWXDISABLE to disable reject file logging.</p> <p>For more information, see the <i>PowerExchange Bulk Data Movement Guide</i>.</p>
STOPONERRORS	0	0 to 2147483647	<p>Available only if the CONFIRMWRITE parameter is set to T, which enables asynchronous with fault tolerance mode.</p> <p>The number of non-fatal errors that are allowed in the write phase before processing is terminated.</p> <p>For more information, see the <i>PowerExchange Bulk Data Movement Guide</i>.</p>
UAP	None	-	<p>For DB2 for i5/OS data sources only. Overrides the fully qualified library and journal name specified in the PowerExchange capture registration.</p> <p>For example:</p> <pre>STQA/NEWJOURNAL</pre>

Parameter	Default	Tokens	Usage
WAITTIME	-	0 2 to 86399 86400	The maximum approximate time, in seconds, to wait for data before returning end-of-file (EOF). The options are: - 0. The EOF is returned when the end of the current log is reached. The end of the log is determined at the start of the extraction process, because the actual end point constantly changes. - 2 to 86399. The number of seconds to wait for data before returning EOF. - 86400. The EOF is never returned. The job waits forever.
XTRASchema	-	Up to 128 characters with no spaces allowed.	Overrides the schema defined in the extraction map.

The following table describes the DBQUAL1 parameter:

Tokens	DBType
i5/OS Library/Filename	CAPX and CAPX/RT
SSID or DBName	DB2, DB2400C, and DB2UDB
Not applicable	DB2390IMG
Not applicable	IMSUNLD
DSN	MSSQL
Not applicable	NRDB and NRDB2
SQL *Net Name	ORACLE

The following table describes the DBQUAL2 parameter:

Tokens	DBType
Application name	CAPX and CAPX/RT
Not applicable	DB2, DB2400C, and DB2UDB
SSID	DB2390IMG
Not applicable	IMSUNLD
Database	MSSQL
Not applicable	NRDB and NRDB2

Tokens	DBType
Not applicable	ORACLE
Any supported data source, such as DB2	Access method for file or database.

PowerExchange ODBC Driver for DB2 for z/OS Bulk Parameters

The following table describes the PowerExchange ODBC driver parameters that you can specify for DB2 for z/OS bulk load processing:

Field	Type	Size	Valid Values	Comments
BULKLOAD	Char	1	Y or N	Enables or disables DB2 bulk load processing. If you specify Y, you can configure the following parameters.
CTLTEMPLATE	Char	8	User-defined	The name of the DB2 bulk load control file template on the target system, which is a member in a PDS defined on the system where the PowerExchange Listener is running. If you specify a CTL template, the LOADOPTIONS parameter is ignored. If you set CTLTEMPLATE to NONE, no DB2 bulk load control file template is specified. Default is DB2LDJCL. You must customize this JCL before using it.
DELETETEMPFILES	Numeric	int	1, 2, 3, or 4	1 = Does not delete temporary files 2 = Deletes temporary files before running the loader 3 = Deletes temporary files after running the loader if the return code=0 Valid if MODETIME=1 4 = Deletes temporary files after running the loader Valid if MODETIME=1
FNAME	Char	27	File on host	Used as the stem for files created on z/OS. The files created are: - FNAME.DTLnnnnn. Data for load. - FNAME.DTLnnnnn.CTL. Load commands. - FNAME.DTLnnnnn.SQL. SQL template - create table. - FNAME.DTLnnnnn.SYSPRINT. Output from loader.
JCLTEMPLATE	Char	8	User-defined	Name of the JCL file template on the target system, which is a member in a PDS defined on the system where the PowerExchange Listener is running. Default is DB2LDJCL. You must customize this JCL before using it.
LOADOPTIONS	-	-	1 or 4	RESUME (1) or REPLACE (4).

Field	Type	Size	Valid Values	Comments
MODETIME	Numeric	int	1, 2, 3, or 4	<p>1 = WAIT Synchronous. Report results at end. Applicable only if MODETYPE=1 or MODETYPE=3 is specified.</p> <p>2 = NOWAIT Asynchronous. Returns OK when job is submitted. Applicable only if MODETYPE=3 or MODETYPE=2 is specified.</p> <p>3 = TIMED Synchronous. Wait a specified number of seconds. Applicable only if MODETYPE=3 is specified.</p> <p>4 = DATAONLY Only creates the data DB2 Loader file. Applies to all mode type values, but typically used with NOSUBMIT, MODETYPE=2.</p>
MODETYPE	Numeric	int	1, 2, or 3	<p>1 = TASK Submitted as a subtask of the PowerExchange Listener.</p> <p>2 = NOSUBMIT Not submitted. Instead, creates the DB2 loader-specific control files and data files, saves them on the target system, and generates an executable JCL deck on disk.</p> <p>3 = JOB Submitted to the internal reader as a job.</p>
PRISPACE	Numeric	int	1-System defined	Amount of primary space for z/OS files.
SECSpace	Numeric	int	1-System defined	Amount of secondary space for z/OS files.
SPACE	Char	1	T = Tracks C = Cylinders	z/OS allocation of space.
TIME	Numeric	int	0 to 99999	<p>A value from 0 to 99999, as follows:</p> <ul style="list-style-type: none"> - 0. No wait. - 99999. Wait. - Any other value. The number of seconds to wait. <p>Only used in conjunction with MODE= (JOB, TIMED).</p>

SQL Escape Sequences

SQL Escape sequences can be used to override PowerExchange defaults.

SQL Escape Sequences with PowerCenter

In the PowerCenter Workflow Manager, you can enter SQL escape sequences for a particular task.

Note: If you enter multiple escape sequences, enclose each one in curly brackets.

For example:

```
{DTLXTYPE=RS}{DTLIMTYPE=BA}{DTLTIMEOUT=60}d8robora1
```

For more information, see *PowerExchange Interfaces for PowerCenter*.

SQL Escape Sequences Available in PowerCenter

The following SQL escape sequences are available:

- [“{DTLADAPREFETCH=Y|N} Escape Sequence” on page 416](#)
- [“{DTLADAPWD= } Escape Sequence” on page 416](#)
- [“{DTLAPP= } Escape Sequence” on page 416](#)
- [“{DTLARRAYSIZE= } Escape Sequence” on page 416](#)
- [“{DTLCONNOVR= } Escape Sequence” on page 416](#)
- [“{DTLCONFWRITE=N|Y|T} Escape Sequence” on page 416](#)
- [“{DTLDATAPWD= } Escape Sequence” on page 417](#)
- [“{DTLDB2DEGREE= } Escape Sequence” on page 417](#)
- [“{DTLDSN= } Escape Sequence” on page 417](#)
- [“{DTLEVENTTB= } Escape Sequence” on page 418](#)
- [“{DTLFILELIST=Y|N} Escape Sequence” on page 418](#)
- [“{DTLIMTYPE=BA|AI|TU} Escape Sequence” on page 418](#)
- [“{DTLIMGOV=Y|N} Escape Sequence” on page 418](#)
- [“{DTLINSMODE=LOAD|UPDATE} Escape Sequence” on page 418](#)
- [“{DTLJRNL= } Escape Sequence” on page 418](#)
- [“{DTLLIBRARYLIST= } Escape Sequence” on page 419](#)
- [“{DTLLOGSID=} Escape Sequence” on page 419](#)
- [“{DTLMODELDCB=} Escape Sequence” on page 419](#)
- [“{DTLNOUPDATECDEP=Y|N} Escape Sequence” on page 419](#)
- [“{DTLORACOLL= } Escape Sequence” on page 419](#)
- [“{DTLORACONN= } Escape Sequence” on page 419](#)
- [“{DTLORAINST= } Escape Sequence” on page 419](#)
- [“{DTLORASchema= } Escape Sequence” on page 420](#)
- [“{DTLOVRDBF= } Escape Sequence” on page 420](#)
- [“{DTLREJECTFILE= } Escape Sequence” on page 420](#)
- [“{DTLSESSID=} Escape Sequence” on page 420](#)
- [“{DTLSTOPONERRORS =} Escape Sequence” on page 420](#)
- [“{DTLSTRIPORDERBY=Y|N} Escape Sequence” on page 420](#)
- [“{DTLTIMEOUT= } Escape Sequence” on page 421](#)
- [“{DTLUDBDB= } Escape Sequence” on page 421](#)

- [“{DTLUPDELSEINS=Y} Escape Sequence” on page 421](#)
- [“{DTLUSECATALOGMETADATA=Y|N} Escape Sequence” on page 421](#)
- [“{DTLWORKERS= } Escape Sequence” on page 421](#)
- [“{DTLXTRASHEMA= } Escape Sequence” on page 421](#)
- [“{DTLXTYPE=RS|SL} Escape Sequence” on page 421](#)

{DTLADAPREFETCH=Y|N} Escape Sequence

Adabas. Enables users to set prefetch for Adabas.

{DTLADAPWD= } Escape Sequence

Adabas only. Enables users to specify an Adabas file password as an escape sequence in ODBC.

{DTLAPP= } Escape Sequence

Enables you to specify the application name that identifies extracts, increasing the ability to identify individual use of a single connection. This is then available to task control.

{DTLARRAYSIZE= } Escape Sequence

Defines the DB2 fetch array size, in number of rows, for DB2 bulk data movement operations that use the DB2 access method. The array size pertains to the DB2 multiple-row FETCH statements that PowerExchange uses to read data from DB2 source tables.

Valid values are from 1 through 5000. Default is 25.

Note: PowerExchange dynamically lowers the array size when all the following conditions are true:

- The database type is DB2.
- The table contains LOB columns.
- The **Array Size** value is greater than 1.
- Row size * **Array Size** is greater than 16000000 bytes.

{DTLCONFWRITE=N|Y|T} Escape Sequence

Overrides the Write Mode.

- N = Confirm Write Off
- Y = Confirm Write On
- T = Asynchronous Write with Fault Tolerance.

{DTLCONNOVR= } Escape Sequence

Overrides the default CAPI connection specified in the DBMOVER configuration file. Specify the value given in the NAME parameter of the required CAPI_CONNECTION statement in the DBMOVER configuration file to point to the relevant data source.

{DTLADAPWD= } Escape Sequence

Where xxxx is a password needed to access the physical data file on a machine. Enables users to specify a file password as an escape sequence in ODBC. This operates the same way as DTLADAPWD and in fact is the preferred keyword with DTLADAPWD being kept as a synonym for backward compatibility. Used with NRDB and NRDB2 access methods.

{DTLDB2DEGREE= } Escape Sequence

DB2 for i5/OS and z/OS only. Sends the SET CURRENT DEGREE SQL command to DB2, specifying whether or not DB2 should use inter-partition parallel processing. Can be any text string up to 30 characters.

The actual effect of each value can be found in the relevant IBM DB2 documentation for the "SET CURRENT DEGREE" and "CHGQRYA DEGREE" commands.

Using DTLDB2DEGREE on DB2 for z/OS

Any text following the = is taken as text to follow a 'SET CURRENT DEGREE ='.

For example, specifying DB2DEGREE=ANY would execute the statement "SET CURRENT DEGREE = 'ANY'" and would set DB2 to be able to use parallel processing.

Using DTLDB2DEGREE on DB2 for i5/OS

Any text following the = is taken as text to follow a 'CHGQRYA DEGREE' statement.

For example, specifying DB2DEGREE=SYSTEM would execute the statement "CHGQRYA DEGREE(*SYSTEM)".

To maintain compatibility with previous releases, if the parameter begins with a digit, it is appended to a "CHGQRYA DEGREE(*NBRTASKS)" statement. A "CHGQRYA DEGREE(*NBRTASKS 1)" statement can be generated with either a DB2DEGREE=1 or DB2DEGREE="*NBRTASKS 1" statement.

{DTLDSN= } Escape Sequence

Enables SQL to override the physical file name coded in the data map. This enables users to use ODBC to read NRDB and NRDB2 data from various files through one ODBC data Source. For ODBC, this is coded anywhere in the SQL as {DTLDSN=xxx}.

```
TABLE="{DTLDSN=filename}NRDBSchema.NRDBMapName.NRDBTableName"  
TABLE="{DTLDSN=filename}DB2Schema.DB2Table"
```

- Filename is any valid filename for the system on which it resides. For NRDB and NRDB2 the filename will override the file name specified in the PowerExchange data map NRDBSchema.NRDBMapName.NRDBTableName.
- For the DB2 bulk load, filename will override the value FName specified in the ODBC setup (ODBC.INI).
- For i5/OS, the fully-qualified library and table name to be used instead of that specified in the PowerExchange change data capture registration. For example:

```
{DTLDSN=STQA/NEWTABLE}
```

If tables exist in multiple libraries you can define this by using an asterisk (*) instead of the library name. Matching tables in all libraries will be used.

For example:

```
{DTLDSN=*/NEWTABLE}
```

{DTLEVENTTB= } Escape Sequence

Activates Event Processing. Supply the name of the extraction map of the Event Table.

{DTLFILELIST=Y|N} Escape Sequence

Flat files, sequential data sets, ESDSs, and tape data sets only. Defines whether the file name for the session represents a data file or a file list file.

- **N** = The file name for the session represents the name of the file from which source data is to be extracted.
- **Y** = The file name for the session does not contain data. Instead, the file name represents a *filelist file* that specifies the names of the files from which source data is to be extracted. The PowerExchange Listener extracts data from all of the files named in the filelist file in the order in which the files are listed.

{DTLIMTYPE=BA|AI|TU} Escape Sequence

Enables users to specify the capture image type when accessing PowerExchange Change Data Capture through ODBC:

- **BA**. Delivers the before image (before the latest change) and the after image (the latest change). The before image is shipped with an action character forcing the deletion of the record. The after image (which includes the change data) is in the format of an Insert.
- **AI**. Delivers only the after image data for the update.
- **TU**. Delivers the data before and after images as they occur. The before image is shipped to enable the application to ensure nothing has changed in that record since the request. The after image comes through as a change (or Update) record. The update is still a full image but the application picking this up will be able to change the target record rather than Delete it and Insert the new one as when using BA.

{DTLIMGOV=Y|N} Escape Sequence

DB2 image copy access method (DB2390IMG) only. If the imagecopy data set is specified, PowerExchange checks that the imagecopy file exists and that a consistent imagecopy is selected. If either of these is not true, for example, the imagecopy file does not exist or it is inconsistent, then the request is rejected and a message is displayed.

This behavior can be overridden if required by supplying the parameter {DTLIMGOV=Y}. This enables inconsistent imagecopies to be read.

{DTLINSMODE=LOAD|UPDATE} Escape Sequence

The purpose of this escape sequence is to enable empty VSAM files to be loaded and updated. LOAD is the default and requires that all the operations be INSERTs. If the input file includes both inserts and then updates to those inserted records specify DTLINSMODE=UPDATE.

{DTLJRNL= } Escape Sequence

i5/OS only. The fully qualified library and journal name that is to be used instead of that specified in the PowerExchange DBMOVER configuration file.

For example:

```
{DTLJRNL=STQA/NEWJOURNAL}
```

{DTLLIBRARYLIST= } Escape Sequence

i5/OS only. Space-delimited list of libraries that PowerExchange will search in order to qualify the first table name (if the table name is unqualified) on a select, insert, delete or update statement.

A maximum of 34 libraries can be specified, each with a maximum length of 10 characters.

Note: If both the library list override (LIBRARYLIST) and the file list override (OVRDBF) are specified, and a table exists in the LIBRARYLIST as well as in the OVRDBF, the OVRDBF will take precedence.

{DTLLOGSID=} Escape Sequence

CA IDMS/DB CDC data sources only. Specifies the location of the CA IDMS/DB logs and the PowerExchange log catalog.

{DTLMODELDCB=} Escape Sequence

SEQ targets only. Represents a z/OS DDNAME that provides the RECFM, LRECL, and DSORG attributes when a file is dynamically allocated.

DTLMODELDCB provides the ability to allocate a new sequential file with the same properties as an existing one that might have been created in an earlier step.

{DTLNOUPDATECDEP=Y|N} Escape Sequence

If set to Y, does not update the PowerExchange CDEP file for the application.

{DTLORACOLL= } Escape Sequence

Oracle CDC only. Enables the PowerExchange Listener to submit multiple simultaneous Oracle Capture processes that can connect to different Oracle instances. Oracle Capture will now use the override to determine to which instance it should connect, as opposed to using the ORACOLL keyword of the CAPI_CONNECTION TYPE=ORCL statement in dbmover.cfg.

This enables the customer to use a single PowerExchange Listener to capture data from as many as 10 Oracle instances simultaneously.

{DTLORACONN= } Escape Sequence

Oracle CDC only. Enables the user to override Oracle connection information for a given Oracle Collection Id. This means that the user can use a single set of registrations to capture data from multiple Oracle instances.

This overrides the fourth value in the ORACLEID statement of the dbmover.cfg file, for example:

```
ORACLEID=(coll_id,oracle_sid,connect_string,cap_connect_string)
```

Used in conjunction with the Oracle Instance.

Users can specify either or both Instance/Connection string. If one of the keywords is not specified, Oracle Capture will pick up the value of the other from dbmover.cfg.

{DTLORAINST= } Escape Sequence

Oracle CDC only. Enables the user to override Oracle instance information for a given Oracle Collection Id. This means that the user can use a single set of registrations to capture data from multiple Oracle instances.

This overrides the second value in the ORACLEID statement of the dbmover.cfg file, for example:

```
ORACLEID=(coll_id,oracle_sid,connect_string,cap_connect_string)
```

Used in conjunction with the Oracle Connection string.

Users can specify either or both Instance/Connection string. If one of the keywords is not specified, Oracle Capture will pick up the value of the other from dbmover.cfg.

{DTLORASchema= } Escape Sequence

Oracle CDC only. Enables users to override the schema name for a group of capture registrations. This means that users can use a single set of registrations to capture data from multiple schemas that exist in a given Oracle instance.

{DTLOVRDBF= } Escape Sequence

i5/OS only. Using the database file override to replace any occurrence of the specified filename, in any SQL statement, with the new library/file/member combination. The override occurs whether the file is qualified or not and includes any files that are qualified using DTLLIBRARYLIST.

Use the following format:

```
from_file/to_library/to_file{/to_member}
```

Where:

- *from_file*. The file to be overridden.
- *to_library*. The new library to use.
- *to_file*. The file in the new library to use.
- *to_member*. Optional. The member in the new library and file to use. *FIRST is used if nothing is specified. In the latter case the member *FIRST will be assumed.

Note: If both the library list override (DTLLIBRARYLIST) and the database file override are specified, and a table exists in the DTLLIBRARYLIST as well as in the DTLOVRDBF, the DTLOVRDBF will take precedence.

{DTLREJECTFILE= } Escape Sequence

Overrides the asynchronous write reject file. For more information about reject files, see *PowerExchange Bulk Data Movement Guide*.

{DTLSESSID=} Escape Sequence

A user supplied string of maximum 8 characters. Used to link a task with statistics written to SMF or file.

{DTLSTOPONERRORS =} Escape Sequence

Overrides the stop on errors counter. The permitted range is 0-2147483647.

{DTLSTRIPORDERBY=Y|N} Escape Sequence

If set to Y strips all following Order By clauses in the SQL. In addition, PowerExchange will treat any "--" characters in the SQL as comment indicators and will strip out the comment indicators and the following SQL text.

{DTLTIMEOUT= } Escape Sequence

Sets the maximum approximate time (in seconds) to wait for data on a queue before returning EOF.

0 indicates EOF should be returned as soon as the end of the current log is reached. The end of the log is determined at the START of the extract process (as the real end is an ever moving target).

86400 indicates that EOF is never returned. The job will wait forever.

{DTLUDBDB= } Escape Sequence

DB2 for Linux, UNIX, and Windows only. This parameter overrides the database connection obtained from the extraction map.

{DTLUPDELSEINS=Y} Escape Sequence

If set to Y, enables the update else insert mode.

{DTLUSECATALOGMETADATA=Y|N} Escape Sequence

DB2 for z/OS. Specify {DTLUSECATALOGMETADATA=Y} to read metadata from the catalog during a DB2 bulk load session, rather than issuing a SELECT statement. When PowerExchange reads the metadata from the catalog, it is not necessary to grant PowerExchange SELECT privileges on the table.

{DTLWORKERS= } Escape Sequence

This escape sequence is deprecated.

{DTLXTRASchema= } Escape Sequence

This parameter overrides the schema defined in the extraction map.

{DTLXTYPE=RS|SL} Escape Sequence

Enables users to specify the capture extract type when accessing PowerExchange Change Data Capture (CDC) through ODBC.

CHAPTER 12

PowerExchange Datatypes and Conversion Matrix

This chapter includes the following topics:

- [PowerExchange Datatype Conversions, 422](#)
- [ODBC and PowerExchange Datatype Equivalencies, 423](#)

PowerExchange Datatype Conversions

PowerExchange supports the conversions listed in the following tables.

An X in a cell in the table indicates that a conversion is supported. Not all systems support 64-bit integers. Where it is not supported, PowerExchange provides an emulation using floating point conversions. This provides precision up to approximately 15 digits, not the full 18 of a 64-bit integer.

The following table lists the supported conversions for the BIN, DATE, DBL, FLT, TIME, TIMESTAMP, and VARBIN datatypes:

From	To BIN	To CHAR	To DATE	To DBL	To FLT	To STRING	To TIME	To TIMESTAMP	To VARBIN	To VARCHAR
BIN	X	X	-	-	-	X	-	-	X	X
DATE	-	X	X	-	-	X	-	X	-	X
DBL	-	X	-	X	-	X	-	-	-	X
FLT	-	X	-	X	X	X	-	-	-	X
TIME	-	X	-	-	-	X	X	X	-	X
TIMESTAMP	-	X	X	-	-	X	X	X	-	X
VARBIN	X	X	-	-	-	X	-	-	X	X

The following table lists the supported conversions for the CHAR, NUMCHAR, NUMxx, PACKED, STRING, UPACKED, UZONED, VARCHAR, and ZONED datatypes:

From	To CHAR	To DBL	To NUMCHAR	To NUMxx	To PACKED UPACKED	To STRING	To VARCHAR	To ZONED UZONED
CHAR	X	X	X	X	X	X	X	X
NUMCHAR	X	X	X	X	X	X	X	X
NUMxx	X	X	X	X	X	X	X	X
PACKED	X	X	X	X	X	X	X	X
STRING	X	X	X	X	X	X	X	X
UPACKED	X	X	X	X	X	X	X	X
UZONED	X	X	X	X	X	X	X	X
VARCHAR	X	X	X	X	X	X	X	X
ZONED	X	X	X	X	X	X	X	X

ODBC and PowerExchange Datatype Equivalencies

The following table shows the ODBC and PowerExchange datatype equivalencies:

ODBC SQL Datatype	ODBC SQL C Datatype	PowerExchange Datatype
SQL_BINARY	-	DTLNET_BIN
SQL_BIT	SQL_C_BIT	DTLNET_BIT
SQL_CHAR	-	DTLNET_CHAR
SQL_DATE	SQL_C_DATE	DTLNET_DATE
SQL_DECIMAL	-	DTLNET_MONEY
SQL_DECIMAL	SQL_C_ULONG	DTLNET_NUM32U
SQL_DECIMAL	-	DTLNET_NUM64
SQL_DECIMAL	-	DTLNET_NUM64U
SQL_DECIMAL	-	DTLNET_NUMCHAR
SQL_DECIMAL	-	DTLNET_PACKED
SQL_DECIMAL	-	DTLNET_UPACKED

ODBC SQL Datatype	ODBC SQL C Datatype	PowerExchange Datatype
SQL_DECIMAL	-	DTLNET_UZONED
SQL_DECIMAL	-	DTLNET_ZONED
SQL_DOUBLE	SQL_C_DOUBLE	DTLNET_DOUBLE
SQL_INTEGER	SQL_C_USHORT	DTLNET_NUM16U
SQL_INTEGER	SQL_C_LONG and SQL_C_SLONG	DTLNET_NUM32
SQL_REAL	SQL_C_FLOAT	DTLNET_FLOAT
SQL_SMALLINT	SQL_C_SHORT and SQL_C_SSHORT	DTLNET_NUM16
SQL_SMALLINT	SQL_C_UTINYINT	DTLNET_NUM8U
SQL_TIME	SQL_C_TIME	DTLNET_TIME
SQL_TIMESTAMP	SQL_C_TIMESTAMP	DTLNET_TIMESTAMP
SQL_TINYINT	SQL_C_TINYINT and SQL_C_STINYINT	DTLNET_NUM8
SQL_VARBINARY	SQL_C_BINARY	DTLNET_VARBIN
SQL_VARCHAR	SQL_C_CHAR	DTLNET_STRING
SQL_VARCHAR	-	DTLNET_VARCHAR

APPENDIX A

DTL__CAPXTIMESTAMP Time Stamps

This appendix includes the following topic:

- [Time Stamps That Are Reported in the DTL__CAPXTIMESTAMP Field by Data Source, 425](#)

Time Stamps That Are Reported in the DTL__CAPXTIMESTAMP Field by Data Source

The time stamp that PowerExchange reports in the generated DTL__CAPXTIMESTAMP field in change records depends on the data source type and on certain parameter settings.

For PowerExchange data sources on z/OS, the TIMESTAMP parameter in the UOWC CAPI_CONNECTION controls the type of time stamp that PowerExchange reports in the DTL__CAPXTIMESTAMP field. If you set the TIMESTAMP parameter to COMMIT, PowerExchange reports the time stamp of the transaction commit on the source for all changes in the transaction. If you use the default parameter value of LOG, PowerExchange retrieves the time stamp from the source database logs. In this case, the time stamp type depends on the source type.

The following table describes the time stamps that PowerExchange reports when you use the default value of LOG for the TIMESTAMP parameter:

Data Source Type	Time Stamp Type
Adabas	The HDDATE time stamp from the PLOG block header, which indicates when the block was written. Note: In Adabas environments with a low level of update activity, the same time stamp might be reported for multiple updates that occurred at different times.
Datacom table-based CDC	The Coordinated Universal Time (UTC) time or local time when the change record was written to the Datacom LXX log. The LOCAL_TIME parameter in the ECCR configuration member, ECCRDCMP, controls whether the UTC or local time is used.
Db2 for i (i5/OS)	An i5/OS journal time stamp that reflects when the change was recorded in the journal.
Db2 for z/OS	The time at which the DB2 ECCR captured the change data record. Each record in a UOW has a different time stamp. Usually, this time stamp is a UTC value that reflects the time zone of the DB2 for z/OS system.

Data Source Type	Time Stamp Type
IDMS	The time at which the change data record was written to the IDMS log file. This time stamp is equivalent to the storeclock (STCK) time stamp. It does not reflect the local time zone.
IMS log-based CDC	The time at which the change was recorded in the IMS logs.
IMS synchronous CDC	The time at which the change occurred.
Batch VSAM and CICS/VSAM	The time at which the change record was captured. Each record in a UOW has a different time stamp. Usually, this time stamp is a UTC value.

For other data sources that do not use the UOWC CAPI_CONNECTION statement, PowerExchange determines the appropriate time stamp to report in the DTL__CAPXTIMESTAMP field. For PowerExchange Express CDC for Oracle sources, the TIME_STAMP_MODE parameter in the OPTIONS statement of the Express CDC configuration file controls the time stamp type.

The following table describes the time stamp types that PowerExchange reports for these data sources:

Data Source Type	Time Stamp Type
Db2 on Linux, UNIX, or Windows	The time stamp of the transaction commit. This time stamp is an ascending virtual time stamp (VTS) of the Db2 system, which usually corresponds to the UTC value.
Microsoft SQL Server	The time at which the change was written to the distribution database.
MySQL	The time stamp of the change event that MySQL recorded in the binary log.
PowerExchange Express CDC for Oracle	<p>The time stamp type is controlled by the TIME_STAMP_MODE parameter setting in the OPTIONS statement of the Express CDC configuration file.</p> <ul style="list-style-type: none"> - If you use the default value of LOGTIME, PowerExchange reports the time stamp of the change on source database, as recorded in the redo logs. This time stamp reflects the local time zone. - If you specify COMMITTIME, PowerExchange reports the time stamp of the transaction commit on the source database. - If you specify BEGINTIME, PowerExchange reports the time stamp of the begin UOW log record.
PostgreSQL	The time of the transaction commit.

APPENDIX B

PowerExchange Glossary

access method

A source-specific program interface that PowerExchange uses to extract bulk or change data from a data source, such as a table, data set, change stream, condense file, or flat file.

active logs

The disk data sets in which the PowerExchange Logger stores captured change data. See also [archive logs on page 428](#).

after image

A representation of a source record or row after an insert or update operation. PowerExchange can save after images as part of the change data stream under certain conditions. See also [before image on page 428](#).

Agent

See [PowerExchange Agent on page 439](#).

APF

See [authorized program facility \(APF\) on page 428](#).

application group

In the PowerExchange Navigator, a set of application names for the extraction processes that run against the data captured at a specific location. An application group is generated when you define a registration group. The application group is used to filter the data in the CDEP file for display in the PowerExchange Navigator.

application name

A unique name that you assign to an extraction process when you are using PowerCenter to extract the change data. In the PowerExchange Navigator, application names are shown under the application group for the location from which the change data is being extracted.

application service

A service that runs on one or more nodes in the Informatica domain. You create and manage application services in Informatica Administrator or through the infacmd command program. Configure each application service based on your environment requirements.

Application Transparent - Transport Layer Security (AT-TLS) file

A policy file that you can define to specify the z/OS jobs that are authorized to use Secure Sockets Layer (SSL) security.

archive logs

In PowerExchange, the backup data sets that the PowerExchange Logger generates on disk or tape to hold change data that is off-loaded from an active log when it becomes full. After the data is off-loaded, the logging of data to the active log can resume.

asynchronous capture

A type of change data capture where the changes are read from database log files instead of directly from the source database records as transactions occur. This method is used for all relational database sources and for log-based capture. See also [synchronous capture on page 444](#).

asynchronous write

A mode of writing data to a database where the client sends blocks of data to the PowerExchange Listener and does not wait for a response before sending another block of data. See also [synchronous write on page 445](#).

authorized program facility (APF)

An IBM facility in a z/OS environment that you can use in PowerExchange to establish authorization for programs to use restricted functions.

batch extraction mode

A method of extracting change data from PowerExchange Condense condense files or PowerExchange Logger for Linux, UNIX, and Windows log files, whereby the extraction runs in batch mode. Only the files that are closed when the extraction job starts are processed. After the extraction job completes reading changes from those condense files or log files, it terminates. See also [continuous extraction mode on page 431](#) and [real-time extraction mode on page 443](#).

Batch Message Processing (BMP)

An IMS batch processing program that has access to online databases and message queues but is started with JCL. The PowerExchange IMS synchronous ECCR can capture change data from BMP sessions.

before image

A representation of a source record or row before the last delete or update operation is processed. PowerExchange can save before images as part of the change data stream under certain conditions. See also [after image on page 427](#).

BMP

See [Batch Message Processing \(BMP\) on page 428](#).

bulk data extraction

The movement of source data in bulk to materialize a target or to synchronize a target data with the source data at a specific point in time.

CAPI

See [consumer API \(CAPI\) on page 430](#).

CAPI connection

The consumer API connection that is defined in the DBMOVER configuration file and that PowerExchange uses to extract captured change data from the change stream that is for a specific source.

CAPTPARM configuration member

The member that defines configuration parameters for PowerExchange Condense on i5/OS and z/OS. On i5/OS, this member resides in the CFG file in the *datilib* library and in the CFGCOND file in the *condlib* library. On z/OS, this member is in the RUNLIB library. Corresponds to the PWXCCL configuration file on Linux, UNIX, and Windows. See also [PowerExchange Condense on page 439](#).

capture catalog table

The table that you create for storing information about DB2 for Linux, UNIX, and Windows tables that are registered for change data capture, including their column definitions and valid log positions. The capture catalog table must reside in the same database as the registered tables. You initialize the capture catalog table by issuing the SNAPSHOT command of the DTLUCUDB utility.

capture directory tables

A set of tables that the DB2 ECCR uses to track information about the DB2 for z/OS tables that are registered for change data capture. These tables are created at PowerExchange installation and must reside on the same DB2 subsystem as the tables registered for change data capture.

capture registration

A named definition of the source for which you want PowerExchange to capture changes. In the PowerExchange Navigator, you define a capture registration under a registration group. The registration addresses a single table or data set and specifies the columns or fields for which to capture changes, the **Condense** option to use, and the registration status. See also [registration group on page 443](#).

CAPX

An access method that PowerExchange uses to access change data in PowerExchange Condense condense files or PowerExchange Logger for Linux, UNIX, and Windows log files when you use batch extraction mode. Also, a type of CAPI_CONNECTION statement in the dbmover configuration file for a relational database source on a Linux, UNIX, or Windows system. This statement specifies parameters for extracting change data from condense files or PowerExchange Logger log files in continuous extraction mode.

CAPXRT

An access method that the PowerExchange Listener uses to access change data from the change stream.

CCT

The file in which PowerExchange stores information about the capture registrations that users have defined in the PowerExchange Navigator.

CDC

See [change data capture \(CDC\) on page 430](#) and [PowerExchange Change Data Capture \(CDC\) on page 439](#).

CDCT file

The file in which PowerExchange Condense on i5/OS or z/OS stores information about each condense file that it generates, or the file in which the PowerExchange Logger for Linux, UNIX, and Windows stores

information about each log file that it generates. This information includes the file name, start and end times, condense type, and number of records.

CDEP file

The control file in which PowerExchange stores application names and information about the extraction processes that have run, including their input and timings. See also [application name on page 427](#).

change data capture (CDC)

In PowerExchange, the process of capturing changes that were made to source databases, tables, and data sets for transmittal to a relational or nonrelational target. See also [data extraction on page 431](#) and [PowerExchange Change Data Capture \(CDC\) on page 439](#).

change stream

The change data that PowerExchange has read from one of the following types of sources:

- PowerExchange Logger for z/OS log files
- PowerExchange Condense condense files on i5/OS or z/OS systems
- RDBMS log files on i5/OS, Linux, UNIX, or Windows systems
- PowerExchange Logger for Linux, UNIX, and Windows log files

checkpoint files

Files that PowerExchange generates to store checkpoint information for PowerExchange Condense on i5/OS or z/OS. This information is used during a warm start to properly resume PowerExchange Condense processing from the point at which it was interrupted.

CmdPrefix

A z/OS command prefix that you use when specifying PowerExchange Agent commands. This prefix is defined in the PowerExchange Agent AGENTCTL configuration parameters.

commit record

A record in the change stream that indicates that the UOW has ended and the changes in the UOW have been committed. For data sources that do not generate a commit record, PowerExchange generates one when it captures the change data. Synonymous with *end-UOW*.

complex table

A table in a data map that is based on more than one record.

Condense

See [PowerExchange Condense on page 439](#).

condense files

The files in which a PowerExchange Condense process on i5/OS or z/OS stores captured change data. See also [PowerExchange Condense on page 439](#).

consumer API (CAPI)

The API PowerExchange uses to extract the captured change data for a specific source type.

continuous extraction mode

A method of extracting change data from PowerExchange Condense condense files or PowerExchange Logger for Linux, UNIX, and Windows log files as the data is being written to the files. The extraction continues to run until you stop it. See also [batch extraction mode on page 428](#) and [real-time extraction mode on page 443](#).

cross-system coupling facility (XCF)

A z/OS component that provides functions to support cooperation between authorized programs running within a sysplex. PowerExchange uses XCF to exchange control information between PowerExchange Change Data Capture components on a single z/OS system or on multiple z/OS systems in a Post-Log Merge environment.

data extraction

The process of extracting captured change data from the change stream so that the data can be moved to a target location. See also [change data capture \(CDC\) on page 430](#) and [PowerExchange Change Data Capture \(CDC\) on page 439](#).

data map

The mapping of nonrelational records to relational tables so that PowerExchange can use the SQL language for relational databases to access the data. Also, a mapping of DB2 fields so that you can split the data in a column that contains multiple fields into separate fields.

data replication

See [replication on page 443](#).

data source

A database unit or file from which PowerExchange can capture change data or move bulk data. Sources include relational database tables, nonrelational database elements, and flat files.

data target

A database unit or file to which the data that PowerExchange captured can be written. PowerExchange works with PowerCenter to load captured data to a target. PowerExchange also provides methods of writing data to nonrelational targets. Targets include relational database tables, nonrelational database elements, and flat files.

datatype conversion

The process of converting data stored in one relational or programmatic datatype to another relational or programmatic datatype.

DBMOVER configuration file

The primary configuration file that the PowerExchange Listener uses in performing PowerExchange functions. The file includes statements that are required for PowerExchange to receive requests from remote systems for processing. It also includes statements that are required to connect to source or target databases.

DTLCCADW utility

A PowerExchange utility that the Adabas ECCR runs transparently to manage the PowerExchange PCAT control file for change data processing. You can run the utility manually to override the default processing if necessary. Also called the Adabas PCAT Utility.

DTLCUIML utility

A PowerExchange utility that you can use to define a marker in the IMS logs for use by the IMS log-based ECCR. When the ECCR encounters a marker, a PowerExchange Logger message is issued that specifies restart tokens for the affected registration tags. You can use these tokens as input to the DTLUAPPL utility to define a start point for extraction processing. The DTLCUIML utility runs as an IMS Batch Message Processing (BMP) job. also called the IMS Log Marker Utility.

DTLINFO utility

A PowerExchange utility that displays the version, release, and build level for PowerExchange or for a specific PowerExchange module. Also called the Release Information Utility.

DTLMSG file

A file from which PowerExchange reads messages for various tasks and processes.

DTLREXE utility

A PowerExchange utility that you can use to submit a z/OS job from a remote z/OS image or from a non-z/OS system or to test the connectivity of a PowerExchange Listener. Also called Remote Execution Utility.

DTLUAPPL utility

A PowerExchange utility that sets or resets extraction restart tokens for all PowerExchange Change Data Capture sources. This utility enables you to add or edit application name entries in the CDEP file, generate restart tokens, and print the restart tokens and application name entries. Also called the Restart Token Utility.

DTLUCBRG utility

A PowerExchange utility that you can use to create capture registrations and extraction maps in batch mode for a set of existing tables or data maps. The utility provides a way to create these items at specified PowerExchange Listener locations based on generic settings. Also called the Batch Registration Utility.

DTLUCDEP utility

A PowerExchange utility that you can use to edit or print the contents of the CDEP file. The file contains information about the extraction processes that have run, including their input and timings. When another extraction process runs, it reads the file to establish a start point. You can use the utility to remove old data from the file to prevent it from becoming too large. Also called the CDEP Utility.

DTLUCSR2 utility

A PowerExchange utility that identifies the locations of IDMS SR3 records after an event such as a database reorganization causes these records to become separated from their corresponding SR2 records. The IDMS ECCR needs the locations of the SR2 records to properly scan the SR2 and SR3 records for change data capture.

DTLUCUDB utility

A PowerExchange utility that you run shortly after installation and prior to creating capture registrations to initialize the PowerExchange capture catalog table for DB2 for Linux, UNIX, and Windows sources. You can also run the utility to generate diagnostic information for troubleshooting data capture problems. Also called DB2 CDC Utility.

DTLULCAT and DTLULOGC utilities

PowerExchange utilities for updating the IDMS Log Catalog, which is used for IDMS log-based change data capture. DTLULCAT prepares the input for DTLULOGC. DTLULOGC populates the Log Catalog with updated information about the logs to process.

DTLURDMO utility

A PowerExchange utility that you can use to copy data maps, capture registrations, and extraction maps to another location. The utility also provides a way to change the attributes of the registrations and data maps, such as the schema name or table name, during the copy process. Also called the Data Map Utility.

DTLUTSK utility

A PowerExchange utility that you can use to list all active PowerExchange tasks, locations, and allocated data sets. You can also stop tasks if necessary. Also called the Task Control Utility.

ECCR

See [Environmental Change Capture Routine \(ECCR\) on page 434](#).

EDMC

A CICS transaction for controlling CICS/VSAM processing. This transaction is used to initialize the ECCR, terminate processing, display the files that are participating in the process, and display a help panel for the ECCR.

EDMLUCTR utility

A PowerExchange utility that you can use to display information about the PowerExchange Logger logs and the captured change data in the logs. This information includes summary information about data sources by registration tag, detail on change records and UOW records, and lists of UOWs that have not yet ended. Also called Log Scan and Print Utility.

EDMMSG data set

A SYSOUT data set on z/OS for messages from the PowerExchange Agent, ECCRs, PowerExchange Logger, Log Read API (LRAPI), and the Log Write API (LWAPI).

EDMNAME

A name that PowerExchange generates and assigns to a registered z/OS source segment, table, database, or data set to identify the change records of interest for CDC processing. For z/OS data sources, the terms EDMNAME and registration tag are equivalent and used interchangeably. You can view EDMNAMEs in the ECCR output in the EDMMSG SYSOUT data set. Alternatively, in the PowerExchange Navigator, look up the equivalent **Tag** names in capture registrations.

EDMSDIR options module

A PowerExchange module that contains the configuration options for the PowerExchange Change Data Capture components on a z/OS system.

EDMUPARM options module

A PowerExchange module that contains parameters for the PowerExchange Logger on a z/OS system.

EDMXLUTL utility

A PowerExchange utility for creating an event marker in the PowerExchange Logger logs on a z/OS system. Also called the Event Marker Utility.

emergency restart data set (ERDS)

A bootstrap data set on a z/OS system that the PowerExchange Logger uses for restarting extractions. The data set contains an inventory of the active and archive log data sets and other control information for the PowerExchange Logger.

encryption mode

The type of data encryption that PowerExchange uses when moving data. You set the encryption mode in the DBMOVER configuration file. These encryption modes are: the PowerExchange algorithm, Digital Encryption Standard (DES), Triple DES, and RC2 by RSA Security. If you use the Secure Sockets Layer (SSL) protocol to secure communications among PowerExchange systems in a network, you can also configure SSL to allow the encryption and decryption of data.

end-UOW

See [commit record on page 430](#).

Environmental Change Capture Routine (ECCR)

A PowerExchange component that captures changes from a source database, table, or data set and writes them to log files. The log files store the changes for future extraction processing. PowerExchange Change Data Capture provides a separate ECCR for each supported source.

ERDS

See [emergency restart data set \(ERDS\) on page 434](#).

error action file

In PowerExchange, a file that contains custom specifications for fault tolerance behavior in error handling. You must specify the location of customized error action files in the PowerExchange DBMOVER configuration file on the target data system.

event marker

In the PowerExchange Logger, a record that represents a specific point in the change stream. Event markers can be used to provide restart points for extractions.

extended recovery facility (XRF)

An IMS facility for minimizing the effects of failures or stoppages in z/OS, VTAM, the host processor, or high-availability applications during sessions between high-availability applications and designated terminals. This facility provides an alternative IMS subsystem that can take over sessions from the failing subsystem.

extraction

See [data extraction on page 431](#).

extraction group

In the PowerExchange Navigator, a set of extraction maps for extracting the data that is captured from a specific data source instance and node location using a specific access method. An extraction group is generated when you define a registration group. The extraction group is used to filter the data in the DTLCAMAP file (on z/OS systems) or the CAMAPS file (on non-z/OS systems) for display in the PowerExchange Navigator.

extraction map

A definition of a change data extraction for a source table or data set. PowerExchange generates an extraction map for each capture registration that you define. You can edit the generated extraction maps to add or remove columns. You can also create additional extraction maps for a capture registration. In the PowerExchange Navigator, extraction maps are shown under the extraction group for the database instance and node location from which the data will be extracted.

extraction mode

A method of extracting change data from the change stream. See also [batch extraction mode on page 428](#), [continuous extraction mode on page 431](#), and [real-time extraction mode on page 443](#).

Flexible Target Key Custom Transformation

A type of transformation that you can define in PowerCenter for a PowerExchange Change Data Capture source. This transformation is useful when the source table contains key columns that are subject to UPDATE SQL commands and the target has key columns that are mapped to those source columns. This transformation generates a pair of delete and insert statements to update the target instead of using the update statement.

full condense

A condense type that is available for PowerExchange Condense on i5/OS or z/OS. With this condense type, change data is accumulated in keyed condense files such that later changes supersede earlier changes. For example, if insert1, update1, and update2 are written to a source field, in that sequence, each change is passed to the condense file but only update2 is available for extraction processing. To use this condense type, you must select **Full** for the **Condense** option when defining capture registrations in the PowerExchange Navigator. This condense type does not maintain transactional consistency. See also [partial condense on page 438](#).

group definition file

See [PowerExchange Condense group definition file on page 439](#) or [PowerExchange Logger group definition file on page 440](#).

group source processing

When using PWXPC connections in PowerCenter workflows, the manner in which PowerExchange reads all data from a physical source in a single pass. For change data, PowerExchange reads the data in the change stream once for all tables in the workflow mapping. For multiple-record VSAM, IMS unload, and sequential data sets, PowerExchange reads the data set once to get data for all record types.

group target processing

When using PWXPC connections in PowerCenter workflows, the manner in which PowerExchange writes all data to a physical target in a single pass. PowerExchange writes the data once for all record types in a multiple-record target.

HOSTENT utility

A PowerExchange utility that you can use to display the TCP/IP host name and address for a system and to diagnose problems related to PowerExchange communication and licensing. Also called TCP/IP Address Reporter utility.

ICU

See [International Component for Unicode \(ICU\) on page 436](#).

in-doubt UOWs

The units of work that have not yet been committed to the source database.

International Component for Unicode (ICU)

A portable set of C/C++ and Java libraries for Unicode support, software internationalization (I18N), and globalization (G11N) code page support, which is provided with PowerExchange.

key range partitioning

A type of PowerCenter pipeline partitioning that you can use for reader partition points in PowerExchange bulk data movement sessions. The PowerCenter Integration Service distributes rows of data based on a port or set of ports that you define as the partition key. For each port, you define a range of values. The PowerCenter Integration Service uses the key and ranges to send rows to the appropriate partition.

listener

See [PowerExchange Listener on page 439](#).

local mode

A method that a process, such as an extraction process, uses to communicate with PowerExchange when the process and PowerExchange are running on the same system. This method does not use IP.

location

In PowerExchange, a user-specified node statement in the DBMOVER configuration file that identifies a PowerExchange Listener on a remote system. The node statement contains a port number and the IP address or host name of the remote source or target system.

log-based capture

An asynchronous method of reading change data from log files.

Logger

See [PowerExchange Logger for Linux, UNIX, and Windows on page 440](#) or [PowerExchange Logger for z/OS on page 440](#).

logger token

See [restart tokens on page 444](#).

logical partition (LPAR)

A logical partition of a computer's resources, such as processors, memory and storage, that is virtualized as a separate computer instance. Each partition can host its own operating system and applications. Multiple

LPARs can run on the same machine. Typically, partitions are used for different functions, such as for test and production environments. A partition communicates with another partition as if the other partition is on a separate machine. LPARs are compatible with multiple operating system types, including z/OS, i5/OS, AIX, and Linux. IBM z/OS systems operate exclusively in LPAR mode.

Log Read API (LRAPI)

A PowerExchange API that connects to the PowerExchange Logger to read the captured change data from the log files.

Log Write API (LWAPI)

A PowerExchange API that connects to the PowerExchange Logger to write the change data that was captured by the ECCR to the log files.

LRAPI

See [Log Read API \(LRAPI\) on page 437](#).

LWAPI

See [Log Write API \(LWAPI\) on page 437](#).

makeconv utility

An open source program provided by the International Component for Unicode (ICU) that converts a source code page definition from a Unified Change Management (UCM) file into a binary CNV file.

materialization

The initial loading of bulk data to a target. After the data loaded, the target is materialized. You can then propagate only the change data to update it.

message destination override

A way for users to suppress PowerExchange messages or to direct these messages to specific locations.

MUFplex

For Datacom sources, a collection of Multi-User Facility (MUF) systems that can share data.

Navigator

See [PowerExchange Navigator on page 440](#).

near real time

A mode in which PowerExchange provides change data as the changes are made.

netport job

In PowerExchange, a JCL stream that you can associate with a port to accommodate special situations. For example, you can use a netport job to access sources through a non-PowerExchange module such as IMS DLIBATCH, alleviate excessive wait times for tape mounts, perform GDG processing, or provide IDMS access with full user ID checking. You link a netport job to a port by setting parameters in the DBMOVER configuration file.

node

In PowerExchange, a remote PowerExchange Listener to which you can connect.

nonrelational source

In PowerExchange, a unit in a database other than a relational database from which PowerExchange can capture changes or move bulk data. Nonrelational sources include IMS segments, IDMS records, Adabas files, VSAM data sets, and flat files. See also [relational source on page 443](#).

nonrelational target

In PowerExchange, a unit in a database other than a relational database to which captured changes or bulk data can be written. Nonrelational targets include IMS segments, IDMS records, Adabas files, VSAM data sets, and flat files. See also [relational target on page 443](#).

normalization

The process of restructuring a data model by reducing its relations to their simplest forms when designing a logical relational database. Normalization helps avoid redundancies and inconsistencies in data.

ODBC

See [Open Database Connectivity \(ODBC\) on page 438](#).

Open Database Connectivity (ODBC)

A standard API that a database application can use to access data in relational and nonrelational database management systems (DBMSs) that use different data storage formats and programming interfaces. PowerExchange provides a thin ODBC driver to connect with PowerCenter so that PowerCenter can read the captured data and write it to various targets. However, to integrate PowerExchange and PowerCenter, it is recommended that you use the PowerExchange Client for PowerCenter (PWXPC) instead of the ODBC driver. See also [PowerExchange Client for PowerCenter \(PWXPC\) on page 439](#).

open UOW

A UOW for which PowerExchange has read the beginning of the UOW from the change stream but has not yet read the commit record.

partial condense

A condense type that is available for PowerExchange Condense and for the PowerExchange Logger for Linux, UNIX, and Windows. With this condense type, changes in UOWs that complete successfully on registered sources are written to PowerExchange Condense condense files or PowerExchange Logger log files in chronological order based on UOW end time. PowerExchange writes all of the changes for the columns of interest, not just the latest changes. To use partial condense, you must select **Part** for the **Condense** option when defining capture registrations in the PowerExchange Navigator. This condense type maintains transactional consistency. See also [full condense on page 435](#).

pass-through partitioning

A type of PowerCenter pipeline partitioning that you can use for PowerExchange bulk data movement sessions. The PowerCenter Integration Service passes all rows from one partition point to the next partition point without redistributing data across partitions. All rows in a partition stay in that partition after crossing a partition point.

personal metadata

The parameters that you define in the PowerExchange Navigator for accessing metadata about the bulk and change data sources that you are interested in. You can also use this information to view data from those sources. These parameters are saved to a personal profile for reuse.

Post-Log Merge

A PowerExchange configuration that uses a job for extracting data that was logged by multiple PowerExchange Loggers. The logged data is merged chronologically for extraction processing. This configuration accommodates sites that use multiple z/OS images and multiple PowerExchange Loggers.

PowerExchange Agent

A PowerExchange component that provides services to other PowerExchange components, including obtaining and managing global queues, creating address spaces, and managing capture registrations.

PowerExchange Call Level Interface (SCLI)

An external interface to PowerExchange.

PowerExchange Change Data Capture (CDC)

The PowerExchange feature for capturing changes to data in a source database.

PowerExchange client

PowerExchange software that enables you to use PowerExchange from a remote computer. The client communicates with a PowerExchange Listener on a different computer.

PowerExchange Client for PowerCenter (PWXPC)

A PowerCenter component that fully integrates PowerCenter with PowerExchange so that PowerCenter can access PowerExchange-controlled data and write it to various targets. PWXPC has several advantages over the alternative ODBC drivers, which PowerExchange also provides for this purpose.

PowerExchange Condense

An optional PowerExchange process for data sources on i5/OS or z/OS. PowerExchange Condense condenses captured change data and stores the data in condense files. The condense files store only the successfully completed UOWs, in chronological order by end time. When an extraction session runs, the data is extracted from the condense files rather than from the change stream.

PowerExchange Condense group definition file

An optional file that defines a group of capture registrations that PowerExchange Condense uses to write change data for registered tables to separate condense files. The use of a separate condense file for each table in a group makes extraction processing of that data more efficient. The group definition file name must be defined in the GROUPDEFS parameter of the dtlca.cfg file.

PowerExchange Listener

The PowerExchange feature that coordinates activities across systems, initiates requests, or processes requests from third-party applications. A PowerExchange Listener must be started if you plan to move data across systems.

PowerExchange Logger for Linux, UNIX, and Windows

An optional PowerExchange process for data sources on Linux, UNIX, and Windows. The PowerExchange Logger captures and stores change data in PowerExchange Logger log files. The log files store only the successfully committed UOWs, in chronological order by end time. When an extraction session runs, the change data is extracted from the PowerExchange Logger log files rather than from the database logs.

PowerExchange Logger for z/OS

A PowerExchange component that is required to perform change data capture for data sources on z/OS. The PowerExchange Logger receives captured change data from an ECCR, creates a persistent copy of the data, and inventories the data. When an extraction session runs, the PowerExchange Logger supplies the data on demand for replication to target databases.

PowerExchange Logger group definition file

An optional file that defines a group of capture registrations that the PowerExchange Logger for Linux, UNIX, and Windows uses to write change data for registered tables to separate log files. The use of a separate log file for each table in a group makes extraction processing of that data more efficient. The file name must be defined in the GROUPDEFS parameter of the pwxcl.cfg file.

PowerExchange Navigator

A PowerExchange Windows client for defining and managing change data and bulk data movements. You must use the PowerExchange Navigator to create capture registrations for data sources. You can also it to create and edit extraction maps, data maps, and personal metadata profiles.

PWXCL configuration file

The file that defines parameters for configuring the PowerExchange Logger for Linux, UNIX, and Windows. Corresponds to the CAPTPARM member on i5/OS and z/OS.

PWXPC

See [PowerExchange Client for PowerCenter \(PWXPC\) on page 439](#).

PWXUCDCT utility

A PowerExchange utility for managing the CDCT file and other files used by the PowerExchange Logger for Linux, UNIX, and Windows. The utility runs on Linux, UNIX, and Windows systems. Use the utility to perform the following tasks:

- Regenerate the CDCT file if it is damaged or deleted.
- Back up, restore, and maintain the CDCT file.
- Delete expired CDCT records and any PowerExchange Logger log files that are no longer referenced by the CDCT file.
- Print reports on the PowerExchange Logger pwxcl.cfg configuration file, CDCT file, checkpoint files, and log files.

Also called the PowerExchange Logger for Linux, UNIX, and Windows Utility.

PWXUCREG utility

A PowerExchange utility that you can use to suspend one or more capture registrations temporarily so that the PowerExchange ECCR does not capture database changes using those registrations during the suspension period. The utility also reactivates the registrations so that change data capture can resume.

This utility runs only on z/OS systems and processes registrations only for the following ECCR types:

- Adabas ECCR
- Datacom table-based ECCR
- IDMS log-based ECCR
- IMS log-based ECCR

Use the utility to perform the following tasks:

- Suspend capture registrations to temporarily stop change capture activity for registered sources during the suspension window.
- Reactivate suspended capture registrations after a suspension to resume change data capture.
- Display the status setting for capture registrations to verify a status change.
- Skip all change records in the change stream that have timestamps earlier than the current system time when starting change capture for a registration that has been activated for the first time from the PowerExchange Navigator.

Also called the Capture Registration Suspend Utility.

PWXUCRGP utility

A PowerExchange utility that reports the capture registration information that is stored in the CCT file, for all registrations or a subset of registrations. Reports can be filtered based on database type, instance, name, registration name, or registration status. You can also report information for a specific capture registration. You can control the report level of detail and format.

The reports contain information that can be used in various situations such as migrating registrations to another system or diagnosing registration-related problems. The report output is written to a text file on the local machine, specified in the syntax.

PWXUDMX utility

A PowerExchange utility that you use to allocate, display, and delete ECSA memory, which holds time stamps of the latest updates to data maps files, and to modify the use counts of a file.

This processing is relevant if you configure data maps caching in multiple jobs mode by defining DMXCACHE_MULTIPLEJOBS=Y in the DBMOVER configuration file.

With the PWXUDMX utility, you can complete the following tasks:

- Allocate less than the 4096 bytes of ECSA memory that the system dynamically allocates.
- Delete ECSA memory.
- Display the contents of ECSA memory with file names and time stamps in legible format.
- Display the contents of ECSA memory in hexadecimal format.
- If a PowerExchange Listener or netport job does not shut down cleanly, decrement the use count of a file.
- Increment the use count of a file.

Also called the PowerExchange ECSA Memory Utility.

PWXUGSK utility

A PowerExchange utility that you use to generate reports about SSL libraries and certificates on z/OS.

With the PWXUGSK utility, you can complete the following tasks:

- Verify that the a specified user ID has the authority to view security certificates for the PowerExchange Listener on z/OS, that the certificates are current and valid, and that the AT-TLS rules can intercept inbound requests, remove the TLS information and send TCP/IP packets to the listener.
- Run a certificate report to view certificate information from a RACF keyring or SAF database.
- Run a cipher report to view the cipher suites that are available to a specified user.
- Run an error-code report to view system SSL errors. These errors can also be found in the TCP/IP JES message log.

PWXUMAP utility

A PowerExchange utility that you can use to generate reports that list PowerExchange data maps and extraction maps. You can use PWXUMAP to provide a snapshot of data maps or extraction maps on your system. You can use the snapshot to verify the data maps and extraction maps that you might want to copy with the DTLURDMO utility, or to provide information to Informatica Global Customer Support.

The utility produces the following types of reports:

- List of data maps and extraction maps
- List of source schemas
- List of map lines in comma-delimited format
- Report of CAPXRT or NRDB metadata
- Summary or detailed report of the currently loaded runtime map

PWXUSSL utility

A PowerExchange utility that you use to generate reports about SSL libraries and certificates on Linux, UNIX, and Windows.

With the PWXUSSL utility, you can complete the following tasks:

- Convert a PKCS12DER format certificate file to the PEM format for use on a Linux, Unix, or Windows machine.
- Verify that the a specified PowerExchange user has the authority to view security certificates for the PowerExchange Listener on Linux, Unix, and Windows, that the certificates are current and valid, and that TCP/IP packets can be sent to the PowerExchange Listener.
- Run a certificate report to view information from a certificate chain file. The report can include multiple certificates in a PEM chain file,
- Run a cipher report to view the cipher suites that are available in the OpenSSL cryptographic library. The report includes the hexadecimal codes that you can use to correlate OpenSSL cipher suites to the AT-TLS cipher suites on z/OS.
- Run a return-code or error-code report to view codes from OpenSSL processing and trouble shoot secure connections.
- Run a configuration report to view the PowerExchange Listener and client SSL configuration and troubleshoot any issues associated with the PowerExchange DBMOVER configuration files.
- Run a version report to view the version of OpenSSL that was used to build the cryptographic library. On Linux and UNIX, the cryptographic library file is named **libpmdcrypto**. On Windows, the file is named **PMLIBEAY32.DLL**. The report includes the date of the build and compiler settings.

Also called the PowerExchange SSL Reporting Utility.

reader partitioning

In PowerExchange bulk data movement sessions, the use of pass-through or key-range partitions at the source qualifier, or reader, partition point to improve session performance. Alternative reader partitioning schemes are available based on the data source type.

real time

In PowerExchange, a change data capture method whereby PowerExchange retrieves the data changes as they are made to a source database.

real-time extraction mode

In PowerExchange, a method of extracting change data from the change stream in near real time, as the changes are made. The extraction continues to run until you stop it. See also [batch extraction mode on page 428](#) and [continuous extraction mode on page 431](#).

register

In PowerExchange, to define the source data resources that you want to participate in data propagation. These definitions are saved to the PowerExchange CCT file. PowerExchange recognizes only the source data that you register.

registration group

In the PowerExchange Navigator, a named set of capture registrations for capturing source data. A registration group defines the data source location, the data source type, the collection identifier, and, optionally, the user ID and password for accessing the source data. It is used to filter the data in the CCT file for display in the PowerExchange Navigator. When you define a registration group, a corresponding application group and extraction group are generated. See also [capture registration on page 429](#).

registration tag

A name that PowerExchange generates and assigns to a registered source segment, table, database, or data set to identify the change records of interest for CDC processing. You can view tag names from the PowerExchange Navigator by opening capture registrations. See also [EDMNAME on page 433](#).

reject file

A PowerExchange file that contains rows of data that are rejected and are not written to the target database or flat file. PowerExchange reject files are created on the target system for asynchronous write operations.

relational source

In PowerExchange, a table in a relational database, such as DB2 or Oracle, from which PowerExchange can capture changes or move bulk data. See also [nonrelational source on page 438](#).

relational target

In PowerExchange, a table in a relational database to which captured changes or bulk data can be written. See also [nonrelational target on page 438](#).

replication

The process of maintaining the same data in multiple locations. PowerExchange in conjunction with PowerCenter replicates data by copying data from a source database and applying that data to a target database, according to your specifications.

repository

In PowerExchange, a collection of all capture registrations for z/OS data sources that are managed by a specific PowerExchange Agent.

restart tokens

Identifiers that determine the start point in the change stream for a PowerExchange extraction. The restart tokens consist of a sequence token (RESTART1) and a logger token (RESTART2). The format and content of these tokens vary by system and source type.

row test

The process of testing a data map, extraction map, or personal metadata profile in the PowerExchange Navigator. The test accesses columns in a data source and displays them in tabular format. The results indicate the change data that PowerExchange will retrieve at runtime or the changes that were retrieved.

schema

A description of the fields or columns in a particular segment, record, or table. This description includes characteristics such as data type, length, and precision.

SCLI

See [PowerExchange Call Level Interface \(SCLI\) on page 439](#).

sequence token

See [restart tokens on page 444](#).

Shadow MUF

For Datacom sources, a Multi-User Facility system within the MUFplex that can take over processing when its partner MUF system fails. See also [MUFplex on page 437](#).

simple table

A table in a data map that is based on a single record.

source

See [data source on page 431](#).

SQL escape sequence

A method for overriding the PowerExchange ODBC settings during execution.

synchronization

In a data replication environment, the process of ensuring, at a single point in time, that a set of data in a target database is an accurate representation of the corresponding set of data in the source database. By moving change data in real time, PowerExchange helps to achieve data synchronization.

synchronous capture

A type of change data capture where the PowerExchange ECCR captures changes from a source as they are made and sends them directly to the PowerExchange Logger logs, where they are available for extraction. This type of capture is used for IMS synchronous, batch VSAM, and CICS/VSAM sources. IMS log-based sources use asynchronous capture. See also [asynchronous capture on page 428](#).

synchronous write

A mode of writing data to a database, table, or file in real time using PowerExchange, where a record or row must be successfully written before the next one is written. See also [asynchronous write on page 428](#).

tag

See [registration tag on page 443](#).

target

See [data target on page 431](#).

unit of work (UOW)

A recoverable sequence of operations that is performed by an application between two points of consistency. A unit of work begins when a transaction starts or at a user-requested syncpoint, and it ends when the transaction ends or at a user-requested syncpoint.

UOW

See [unit of work \(UOW\) on page 445](#).

UOWC

An acronym for the UOW Cleanser and also a type of CAPI connection statement. You can define a CAPI UOWC connection statement in the DBMOVER configuration file to specify parameters for the UOW Cleanser when capturing changes from z/OS, DB2 for i5/OS, and Oracle sources.

UOW Cleanser

A PowerExchange component that reconstructs units of work (UOWs) from a change stream into complete and consecutive UOWs that are in chronological order by end time.

UOW spill file

A file in which the UOW Cleanser stores changes for a UOW that it is processing when memory cache is insufficient to hold the entire UOW.

writer partitioning

In PowerExchange bulk data movement sessions, the use of pass-through partitions at the target instance, or writer, partition point to improve session performance. Writer partitions process SQL inserts concurrently and write them to VSAM or sequential file targets.

XCF

See [cross-system coupling facility \(XCF\) on page 431](#).

XRF

See [cross-system coupling facility \(XCF\) on page 431](#).

INDEX

{DTLUSECATALOGMETADATA [421](#)

A

ABEND_SW statement
 DBMOVER configuration file [50](#)
accessing
 multibyte metadata with ODBC [401](#)
ADA_L3_ALLOW statement
 DBMOVER configuration file [51](#)
ADA_TIMEZONE_CHECK statement
 DBMOVER configuration file [52](#)
Adabas
 cipher codes [181](#)
 DBMOVER statements [35](#)
 security [274](#)
ADABAS_DEFAULT statement
 DBMOVER configuration file [52](#)
ADABAS_MU_SEARCH statement
 DBMOVER configuration file [53](#)
ADABAS_PREFIX statement
 DBMOVER configuration file [54](#)
ADABASCODEPAGE statement
 DBMOVER configuration file [54](#)
ADAOPT statement
 DBMOVER configuration file [55](#)
ADAPREFETCH statement
 DBMOVER configuration file [56](#)
ADASTATS statement
 DBMOVER configuration file [56](#)
ADAUSER statement
 DBMOVER configuration file [57](#)
adding
 custom ICU code pages [357](#)
 user-defined ICU-compatible code pages [356](#)
AES
 encryption [321](#)
alternative log files [219](#)
APPBUFSIZE statement
 DBMOVER configuration file [57](#)
APPBUFSIZEDYN statement
 DBMOVER configuration file [58](#)
AS/400 Properties tab
 in PowerExchange Data Source wizard [392](#)
AS400EVENTMSGQ statement
 DBMOVER configuration file [58](#)
AS400USRJRNCODE statement
 DBMOVER configuration file [59](#)
AS400USRMSGQ statement
 DBMOVER configuration file [60](#)
AS4J CAPL_CONNECTION parameters
 parameters and syntax [62](#)
AT-TLS
 adding a rule [305](#)
 overview [305](#)

AT-TLS (*continued*)
 policy file for PowerExchange [307](#)
 setting up infrastructure [305](#)
authenticating
 user credentials (Linux and UNIX) [262](#)
 user credentials (Windows) [262](#)
AUTHGROUP statement
 sign-on file [295](#)
authorizing
 users to run infacmd pwx commands [171](#), [251](#)
 users to run infacmd pwx commands (Linux, UNIX, and Windows) [261](#)
 users to run PowerExchange Listener commands (i5/OS) [258](#)
 users to run PowerExchange Listener commands (Windows) [261](#)
 users to run pwxcmd commands [171](#), [251](#)
 users to run pwxcmd commands (i5/OS) [259](#)
 users to run pwxcmd commands (Linux, UNIX, and Windows) [261](#)
 users to run pwxcmd commands (z/OS) [278](#)

B

BS statement
 DBMOVER configuration file [61](#)

C

CA certificate, creating [310](#)
CA IDMS/DB statement
 DBMOVER configuration file [36](#)
CAPI connection statements
 AS4J parameters [62](#)
 CAPX parameters [67](#)
 LRAP parameters [69](#)
 MSQL parameters [71](#)
 multiple [204](#)
 MYSQL parameters [77](#)
 ORAD parameters [80](#)
 overview [204](#)
 PostgreSQL parameters [82](#)
 UDB parameters [85](#)
 UOWC parameters [89](#)
CAPI statements
 order of precedence [205](#)
CAPI_CONN_NAME statement
 DBMOVER configuration file [61](#)
CAPI_CONNECTION - AS4J statement
 DBMOVER configuration file [62](#)
CAPI_CONNECTION - CAPX statement
 DBMOVER configuration file [67](#)
CAPI_CONNECTION - LRAP statement
 DBMOVER configuration file [69](#)
CAPI_CONNECTION - MSQL statement
 DBMOVER configuration file [71](#)

- CAPI_CONNECTION - MYSQL statement
 - DBMOVER configuration file [77](#)
- CAPI_CONNECTION - ORAD statement
 - DBMOVER configuration file [80](#)
- CAPI_CONNECTION - PG statement
 - DBMOVER configuration file [82](#)
- CAPI_CONNECTION - UDB statement
 - DBMOVER configuration file [85](#)
- CAPI_CONNECTION - UOWC statement
 - DBMOVER configuration file [89](#)
- CAPI_SRC_DFLT statement
 - DBMOVER configuration file [94](#)
- CAPT_PATH statement
 - DBMOVER configuration file [96](#)
- CAPT_XTRA statement
 - DBMOVER configuration file [97](#)
- capture registrations
 - data set-level security [269](#)
 - granting access to [271](#)
 - RACF class security [270](#)
 - resource profiles [270](#)
 - security example [272](#)
 - security on z/OS [269](#)
- CAPX CAPI_CONNECTION parameters
 - parameters and syntax [67](#)
- CAPX Properties tab
 - in PowerExchange Data Source wizard [393](#)
- CAPXRT Properties tab
 - in PowerExchange Data Source wizard [395](#)
- certificate, creating [310](#), [311](#)
- cipher suites, FIPS 140-2 compliant [302](#)
- CMDNODE statement
 - DBMOVER configuration file [97](#), [206](#)
- code pages
 - custom ICU [357](#)
 - user-defined ICU-compatible [356](#)
- CODEPAGE statement
 - DBMOVER configuration file [98](#)
- COLON statement
 - DBMOVER configuration file [99](#)
- components
 - PowerExchange [16](#)
- COMPRESS statement
 - DBMOVER configuration file [99](#)
- configuring
 - PowerExchange processes to receive pwxcmd commands [206](#)
 - SSL client on Windows, UNIX, or Linux [318](#)
 - SSL server on Windows, UNIX, or Linux [317](#)
 - SSL server on z/OS [309](#)
- CONSOLE_CODEPAGE statement
 - DBMOVER configuration file [100](#)
- CONSOLE_MSG statement
 - DBMOVER configuration file [100](#)
- CONSOLE_TRACE statement
 - DBMOVER configuration file [101](#)
- CONVCHAR statement
 - DBMOVER configuration file [101](#)
- CPX_DIR statement
 - DBMOVER configuration file [102](#)
- creating
 - ODBC data source (UNIX) [399](#)
 - ODBC data source (Windows) [384](#)
- CREDENTIALS_CASE statement
 - DBMOVER configuration file [102](#)
- custom ICU code pages
 - adding [357](#)

D

- data map
 - security [273](#)
- data set access requirements
 - for PowerExchange jobs [265](#)
- data set-level security
 - for capture registrations [269](#)
 - for extraction maps [269](#)
- Datacom
 - security [275](#)
- DATAMAP_SERVER statement
 - DBMOVER configuration file [103](#)
- datatype conversions
 - matrix [422](#)
 - overview [422](#)
- DATERANGE statement
 - DBMOVER configuration file [104](#)
- DB2 for i5/OS DBMOVER statements
 - DBMOVER configuration file [37](#)
- DB2 for Linux, UNIX, and Windows statements
 - DBMOVER configuration file [38](#)
- DB2 for z/OS
 - security [276](#)
- DB2 for z/OS DBMOVER statements
 - DBMOVER configuration file [38](#)
- DB2_BIN_AS_CHAR statement
 - DBMOVER configuration file [105](#)
- DB2_BIN_CODEPAGE statement
 - DBMOVER configuration file [105](#)
- DB2_ERRORFILE statement
 - DBMOVER configuration file [106](#)
- DB2/S390 Properties tab
 - in PowerExchange Data Source wizard [397](#)
- DB2CODEPAGE statement
 - DBMOVER configuration file [106](#)
- DB2DEF_ENCODING statement
 - DBMOVER configuration file [110](#)
- DB2ID statement
 - DBMOVER configuration file [111](#)
- DB2PLAN statement
 - DBMOVER configuration file [112](#)
- DBMOVER configuration file
 - examples [207](#)
 - overview [19](#)
- DBMOVER statements
 - ABEND_SW [50](#)
 - ADA_L3_ALLOW [51](#)
 - ADA_TIMEZONE_CHECK [52](#)
 - ADABAS_DEFAULT [52](#)
 - ADABAS_MU_SEARCH [53](#)
 - ADABAS_PREFIX [54](#)
 - ADABASCODEPAGE [54](#)
 - ADAOPT [55](#)
 - ADAPREFETCH [56](#)
 - ADASTATS [56](#)
 - ADAUSER [57](#)
 - APPBUFSIZE [57](#)
 - APPBUFSIZEDYN [58](#)
 - AS400EVENTMSGQ [58](#)
 - AS400USRJRNCODE [59](#)
 - AS400USRMSGQ [60](#)
 - BS [61](#)
 - CAPI connection [204](#)
 - CAPI_CONN_NAME [61](#)
 - CAPI_CONNECTION - AS4J [62](#)
 - CAPI_CONNECTION - CAPX [67](#)
 - CAPI_CONNECTION - LRAP [69](#)

DBMOVER statements (*continued*)

CAPI_CONNECTION - MSQ [71](#)
CAPI_CONNECTION - MYSQL [77](#)
CAPI_CONNECTION - ORAD [80](#)
CAPI_CONNECTION - PG [82](#)
CAPI_CONNECTION - UDB [85](#)
CAPI_CONNECTION - UOWC [89](#)
CAPI_SRC_DFLT [94](#)
CAPT_PATH [96](#)
CAPT_XTRA [97](#)
CMDNODE [97](#), [206](#)
CODEPAGE [98](#)
COLON [99](#)
COMPRESS [99](#)
CONSOLE_CODEPAGE [100](#)
CONSOLE_MSG [100](#)
CONSOLE_TRACE [101](#)
CONVCHAR [101](#)
CPX_DIR [102](#)
CREDENTIALS_CASE [102](#)
DATAMAP_SERVER [103](#)
DATERANGE [104](#)
DB2_BIN_AS_CHAR [105](#)
DB2_BIN_CODEPAGE [105](#)
DB2_ERRORFILE [106](#)
DB2CODEPAGE [106](#)
DB2DEF_ENCODING [110](#)
DB2ID [111](#)
DB2PLAN [112](#)
DECPOINT [112](#)
DEFAULTCHAR [113](#)
DEFAULTDATE [113](#)
DISABLE_PARTITIONS [113](#)
DISP [114](#)
DM_RESOURCE [115](#)
DM_SUBTASK [115](#)
DMX_DIR [118](#)
DMXCACHE_DELETEECSA [116](#)
DMXCACHE_MAX_MEMORY_MB [117](#)
DMXCACHE_MULTIPLEJOBS [117](#)
DTLMSG_CODEPAGE [119](#)
ENABLE_AUTOCOMMIT [120](#)
ENCRYPT [120](#), [321](#)
ENCRYPTLEVEL [121](#), [321](#)
ENQMAJORNAME [122](#)
ENQSYSTEMS [122](#)
ERRROWNOTFOUND [123](#)
examples [207](#)
EXT_CP_SUPPT [123](#)
FILEMAPPED_MEMORY_DIR [124](#)
for Adabas data sources [35](#)
for CA IDMS/DB data sources [36](#)
for DB2 for i5/OS data sources [37](#)
for DB2 for Linux, UNIX, and Windows data sources [38](#)
for DB2 for z/OS data sources [38](#)
for IMS data sources [39](#)
for Linux, UNIX, and Windows data sources [44](#)
for Microsoft SQL Server data sources [39](#)
for MySQL data sources [40](#)
for nonrelational data sources [40](#)
for Oracle data sources [41](#)
for PostgreSQL data sources [42](#)
for the i5/OS operating system [43](#)
for the z/OS operating system [46](#)
for VSAM data sources [42](#)
GDGLOCATE [125](#), [216](#)
GSBUFSIZE [126](#)
HOSTNAME [126](#)

DBMOVER statements (*continued*)

ICUALIAS [127](#)
ICUCNVPROPERTY [127](#)
ICUCONVERTER [130](#)
ICUDATADIR [131](#)
IMBSBDS [132](#)
IMSID [133](#)
JOBCLASS [134](#)
LDAP_BASE [134](#)
LDAP_BIND_DN [134](#)
LDAP_BIND_EPWD [135](#)
LDAP_BIND_PWD [136](#)
LDAP_BIND_TIMEOUT [136](#)
LDAP_FILTER [137](#)
LDAP_HOST [137](#)
LDAP_LOGIN_ATTRIBUTE [138](#)
LDAP_OPENSSL [138](#)
LDAP_PORT [139](#)
LDAP_SASL_MECH [140](#)
LDAP_SCOPE [140](#)
LDAP_SEARCH_TIMEOUT [141](#)
LDAP_TLS [141](#)
LISTENER [142](#), [206](#)
LOADCTFILE [144](#)
LOADJOBFILE [145](#)
LOG_CODEPAGE [145](#)
LOG_LINE_LIMIT [146](#)
LOGPATH [146](#)
LOGSID [147](#)
LOWVALUES [147](#)
LRECL [148](#)
LU00FILE [148](#)
MAXTASKS [149](#)
MSGPREFIX [150](#)
MSGPREFIX-HYPHEN [150](#)
MSS_ERRORFILE [151](#)
MSSQL_SERVER_CONNECT_TIMEOUT [149](#)
multiple CAPI connections [204](#)
MVSDB2AF [151](#)
NEGSIGN [152](#)
NETPORT [152](#)
NODE [155](#)
NOGETHOSTBYNAME [157](#)
notational conventions [50](#)
NRDB_WRITE_CHAR_NULL_FILL [157](#)
NRDB_WRITE_NUM_NULL_FILL [157](#)
NUMERICSIGN [158](#)
ODBASUPP [158](#)
ORA_ERRORFILE [159](#)
ORACLE_CAPTURE_TYPE statement [159](#)
ORACLE_UNHANDLED_NUMASCHAR statement [160](#)
ORACLECODEPAGE [161](#)
ORACLEID [162](#)
OUSP [164](#)
PC_AUTH [164](#)
PIPE [165](#)
POLLTIME [165](#)
PRE861_COMPAT [166](#)
PRGIND [166](#)
PRGINT [167](#)
PWXSOMAXCONN [167](#)
RACF_CLASS [168](#)
RDBMSINSRTDFLT [168](#)
RECFM [169](#)
REJECT_FILE_DELIMITER [169](#)
RELEASE [170](#)
RMTRDBDIRE [170](#)
RMTSYSNAME [170](#)

DBMOVER statements (*continued*)

- SECURITY [171](#), [251](#), [269](#)
- SECURITY_MSGSUPP [175](#)
- SERVICE_TIMEOUT [175](#)
- SESSID [176](#)
- SHOW_THREAD_PERF [176](#)
- source-specific CAPI_CONNECTION [205](#)
- SPACE [177](#)
- SSL [177](#)
- SSL_ALLOW_SELF_SIGNED [178](#)
- SSL_CIPHER_LIST [179](#)
- SSL_CONTEXT_METHOD [180](#)
- SSL_REQ_CLNT_CERT [180](#)
- SSL_REQ_SRVR_CERT [181](#)
- START_UP_USER_EXIT [181](#)
- STATS [182](#)
- SUBMITTIMEOUT [184](#)
- summary [19](#)
- summary of all statements [20](#)
- summary of operating system-specific [42](#)
- SUP_FUNC [184](#)
- SUP_REQUEST [185](#)
- SUP_SSNAME [185](#)
- SUP_SSTYPE [185](#)
- SUPPRESS_DATA_LOGGING [186](#)
- SVCNODE [186](#), [206](#)
- syntax rules [49](#)
- SYSOUT_TIMESTAMP [188](#)
- TAPEWAIT [188](#)
- TCPIP_ACCEPT_NONBLOCK [188](#)
- TCPIP_DIAGNOSTICS_TRACE [189](#)
- TCPIP_DIAGNOSTICS_TRACE_SZ [189](#)
- TCPIP_SHOW_POOLING [190](#)
- TCPIPBUFFSIZE [191](#)
- TCPIPVER [191](#)
- TEMPHLQ [191](#)
- TEXT_EOF_FOR_BINARY [192](#)
- TIMEZONE [192](#)
- TRACE [193](#)
- TRACING [193](#)
- UNIT [199](#)
- USE_DB_AUTH [199](#)
- USE_TYPE1_FALLBACKS [200](#)
- USESUP [200](#)
- VOLSER [201](#)
- VSAM [201](#)
- WAITDSN [202](#)
- WRT_ERROR_HANDLING [203](#)
- DECPOINT statement
 - DBMOVER configuration file [112](#)
- DEFAULTCHAR statement
 - DBMOVER configuration file [113](#)
- DEFAULTDATE statement
 - DBMOVER configuration file [113](#)
- DISABLE_PARTITIONS statement
 - DBMOVER configuration file [113](#)
- DISP statement
 - DBMOVER configuration file [114](#)
- DM_RESOURCE statement
 - DBMOVER configuration file [115](#)
- DM_SUBTASK statement
 - DBMOVER configuration file [115](#)
- DMX_DIR statement
 - DBMOVER configuration file [118](#)
- DMXCACHE_DELETEECDSA statement
 - DBMOVER configuration file [116](#)
- DMXCACHE_MAX_MEMORY_MB statement
 - DBMOVER configuration file [117](#)

DMXCACHE_MULTIPLEJOBS statement

- DBMOVER configuration file [117](#)
- DTL__CAPXTIMESTAMP field
 - types of reported time stamps by data source [425](#)
- DTLADAPREFETCH [416](#)
- DTLADAPWD [416](#)
- DTLAMCPR DD name [269](#)
- DTLAPP [416](#)
- DTLCAMAP DD name [269](#)
- DTLCONFWRITE [416](#)
- DTLCONNOVR [416](#)
- DTLDATAPWD [417](#)
- DTLDB2DEGREE [417](#)
- DTLDESCRIBE
 - COLUMNS [332](#), [334](#)
 - FKEYS [339](#)
 - metadata syntax [326](#)
 - Metadata Syntax [326](#)
 - PKEYS [337](#)
 - PROCEDURECOLUMNS [330](#), [331](#)
 - PROCEDURES [329](#), [330](#)
 - Qualifiers [342](#)
 - RECORDS [335](#), [336](#)
 - SCHEMAS [341](#)
 - TABLES [326](#), [328](#)
- DTLMSG_CODEPAGE statement
 - DBMOVER configuration file [119](#)
- DTLMSGO file
 - overview [224](#)
- DTLMSGO statements
 - overview [223](#)

E

- ENABLE_AUTOCOMMIT statement
 - DBMOVER configuration file [120](#)
- ENCRYPT statement
 - DBMOVER configuration file [120](#), [321](#)
- encryption
 - AES [321](#)
 - overview [320](#)
 - PowerExchange [321](#)
 - setting defaults in the DBMOVER configuration file [321](#)
- ENCRYPTLEVEL statement
 - DBMOVER configuration file [121](#), [321](#)
- ENQMAJORNAME statement
 - DBMOVER configuration file [122](#)
- ENQSYSTEMS statement
 - DBMOVER configuration file [122](#)
- ERRRROWNFOUND statement
 - DBMOVER configuration file [123](#)
- EXT_CP_SUPPT statement
 - DBMOVER configuration file [123](#)
- extraction maps
 - data set-level security [269](#)
 - granting access to [271](#)
 - RACF class security [270](#)
 - resource profiles [270](#)
 - security example [272](#)
 - security on z/OS [269](#)

F

- FILEMAPPED_MEMORY_DIR statement
 - DBMOVER configuration file [124](#)
- FIPS 140-2 compliance [301](#)

G

- GDGLOCATE statement
 - DBMOVER configuration file [125](#), [216](#)
- General tab
 - in PowerExchange Data Source wizard [389](#)
- GSBUFSIZE statement
 - DBMOVER configuration file [126](#)

H

- HOSTNAME statement
 - DBMOVER configuration file [126](#)

I

- i5/OS
 - security [255](#)
- i5/OS DBMOVER statements
 - DBMOVER configuration file [43](#)
- i5/OS security
 - resource access requirements [256](#)
- ICU-compatible code pages
 - adding [356](#)
- ICUALIAS statement
 - DBMOVER configuration file [127](#)
- ICUCNVPROPERTY statement
 - DBMOVER configuration file [127](#)
- ICUCONVERTER statement
 - DBMOVER configuration file [130](#)
- ICUDATADIR statement
 - DBMOVER configuration file [131](#)
- IDMS
 - netport jobs [210](#)
 - Netport jobs [209](#)
- IMS
 - Netport jobs [209](#)
 - security [277](#)
- IMS call considerations
 - nonrelational SQL [326](#)
- IMS DBMOVER statements
 - DBMOVER configuration file [39](#)
- IMBSBDS statement
 - DBMOVER configuration file [132](#)
- IMSID statement
 - DBMOVER configuration file [133](#)
- infacmd pwx commands
 - authorizing users to run [171](#), [251](#)
 - authorizing users to run (Linux, UNIX, and Windows) [261](#)
 - setting up command-handling service for [186](#)
 - sign-on file examples [299](#)
- infacmd pwx program
 - security for [262](#)
 - security for (Linux, UNIX, and Windows) [260](#)
- International Components for Unicode (ICU) [346](#)

J

- JOBCLASS statement
 - DBMOVER configuration file [134](#)

L

- LDAP_BASE statement
 - DBMOVER configuration file [134](#)
- LDAP_BIND_DN statement
 - DBMOVER configuration file [134](#)
- LDAP_BIND_EPWD statement
 - DBMOVER configuration file [135](#)
- LDAP_BIND_PWD statement
 - DBMOVER configuration file [136](#)
- LDAP_BIND_TIMEOUT statement
 - DBMOVER configuration file [136](#)
- LDAP_FILTER statement
 - DBMOVER configuration file [137](#)
- LDAP_HOST statement
 - DBMOVER configuration file [137](#)
- LDAP_LOGIN_ATTRIBUTE statement
 - DBMOVER configuration file [138](#)
- LDAP_OPENSSL statement
 - DBMOVER configuration file [138](#)
- LDAP_PORT statement
 - DBMOVER configuration file [139](#)
- LDAP_SASL_MECH statement
 - DBMOVER configuration file [140](#)
- LDAP_SCOPE statement
 - DBMOVER configuration file [140](#)
- LDAP_SEARCH_TIMEOUT statement
 - DBMOVER configuration file [141](#)
- LDAP_TLS statement
 - DBMOVER configuration file [141](#)
- Linux
 - security [260](#)
- Linux, UNIX, and Windows DBMOVER statements
 - DBMOVER configuration file [44](#)
- LISTENER statement
 - DBMOVER configuration file [142](#), [206](#)
- LOADCTFILE statement
 - DBMOVER configuration file [144](#)
- LOADJOBFILE statement
 - DBMOVER configuration file [145](#)
- LOG_CODEPAGE statement
 - DBMOVER configuration file [145](#)
- LOG_LINE_LIMIT statement
 - DBMOVER configuration file [146](#)
- LOGPATH statement
 - DBMOVER configuration file [146](#)
- LOGSID statement
 - DBMOVER configuration file [147](#)
- LOWVALUES statement
 - DBMOVER configuration file [147](#)
- LRAP CAPI_CONNECTION parameters
 - parameters and syntax [69](#)
- LRECL statement
 - DBMOVER configuration file [148](#)
- LU00FILE statement
 - DBMOVER configuration file [148](#)

M

- makeconv utility [356](#)
- MAXTASKS statement
 - DBMOVER configuration file [149](#)
- metadata syntax
 - DTLDESCRIBE [326](#)
- Microsoft SQL Server DBMOVER statements
 - DBMOVER configuration file [39](#)
- modes of encryption [320](#)

- monitoring PowerExchange
 - SMF statistics logging and reporting [227](#)
- MSGPREFIX statement
 - DBMOVER configuration file [150](#)
- MSGPREFIX-HYPHEN statement
 - DBMOVER configuration file [150](#)
- MSQL CAPI_CONNECTION parameters
 - parameters and syntax [71](#)
- MSS_ERRORFILE statement
 - DBMOVER configuration file [151](#)
- MSSQL_SERVER_CONNECT_TIMEOUT
 - DBMOVER configuration file [149](#)
- MVSDDB2AF statement
 - DBMOVER configuration file [151](#)
- MYSQL CAPI_CONNECTION parameters
 - parameters and syntax [77](#)
- MySQL DBMOVER statements
 - DBMOVER configuration file [40](#)

N

- NEGSIGN statement
 - DBMOVER configuration file [152](#)
- netport jobs
 - substitution variables in netport JCL [212](#)
- Netport jobs
 - configuration [210](#)
 - IDMS [209](#)
 - reasons for using [209](#)
 - sample jobs [210](#)
- NETPORT statement
 - DBMOVER configuration file [152](#)
- NODE statement
 - DBMOVER configuration file [155](#)
- NOGETHOSTBYNAME statement
 - DBMOVER configuration file [157](#)
- nonrelational DBMOVER statements
 - DBMOVER configuration file [40](#)
- nonrelational SQL
 - IMS call considerations [326](#)
 - overview [322](#)
 - supported syntax [323](#)
- NRDB_WRITE_CHAR_NULL_FILL statement
 - DBMOVER configuration file [157](#)
- NRDB_WRITE_NUM_NULL_FILL statement
 - DBMOVER configuration file [157](#)
- NUMERICSIGN statement
 - DBMOVER configuration file [158](#)

O

- ODBASUPP statement
 - DBMOVER configuration file [158](#)
- ODBC
 - CAPXRT Properties tab in PowerExchange Data Source wizard [395](#)
 - DB2/S390 Bulk Load Properties tab in PowerExchange Data Source wizard [397](#)
 - General tab in PowerExchange Data Source wizard [389](#)
 - PowerExchange Data Source tab [386](#)
 - S/400 Properties tab in PowerExchange Data Source wizard [392](#)
- ODBC data source
 - creating (UNIX) [399](#)
 - creating on Windows [384](#)
- ODBC driver
 - PowerExchange [386](#)

- ODBC SQL C datatypes
 - SQL_C_BINARY [423](#)
 - SQL_C_BIT [423](#)
 - SQL_C_CHAR [423](#)
 - SQL_C_DATE [423](#)
 - SQL_C_DOUBLE [423](#)
 - SQL_C_FLOAT [423](#)
 - SQL_C_LONG [423](#)
 - SQL_C_SHORT [423](#)
 - SQL_C_SLONG [423](#)
 - SQL_C_SSHORT [423](#)
 - SQL_C_STINYINT [423](#)
 - SQL_C_TIME [423](#)
 - SQL_C_TIMESTAMP [423](#)
 - SQL_C_TINYINT [423](#)
 - SQL_C_ULONG [423](#)
 - SQL_C_USHORT [423](#)
 - SQL_C_UTINYINT [423](#)
- ODBC SQL datatypes
 - SQL_BINARY [423](#)
 - SQL_BIT [423](#)
 - SQL_CHAR [423](#)
 - SQL_DATE [423](#)
 - SQL_DECIMAL [423](#)
 - SQL_DOUBLE [423](#)
 - SQL_INTEGER [423](#)
 - SQL_REAL [423](#)
 - SQL_SMALLINT [423](#)
 - SQL_TIME [423](#)
 - SQL_TIMESTAMP [423](#)
 - SQL_TINYINT [423](#)
 - SQL_VARBINARY [423](#)
 - SQL_VARCHAR [423](#)
- OpenSSL [310](#)
- operating system authentication
 - for PowerExchange [171](#), [251](#)
- operating system-specific DBMOVER statements
 - DBMOVER configuration file [42](#)
- ORA_ERRORFILE statement
 - DBMOVER configuration file [159](#)
- Oracle DBMOVER statements
 - DBMOVER configuration file [41](#)
- ORACLE_CAPTURE_TYPE statement
 - DBMOVER configuration file [159](#)
- ORACLE_UNHANDLED_NUMASCHAR statement
 - DBMOVER configuration file [160](#)
- ORACLECODEPAGE
 - and globalization [354](#)
- ORACLECODEPAGE statement
 - DBMOVER configuration file [161](#)
- ORACLEID statement
 - DBMOVER configuration file [162](#)
- ORAD CAPI_CONNECTION parameters
 - parameters and syntax [80](#)
- order of precedence
 - CAPI statements [205](#)
- OUSP statement
 - DBMOVER configuration file [164](#)
- overview
 - PowerExchange [16](#)
 - PowerExchange Agent [16](#)
 - PowerExchange Condense [16](#)
 - PowerExchange ECCR [16](#)
 - PowerExchange Listener [16](#)
 - PowerExchange Logger for Linux, UNIX, and Windows [16](#)
 - PowerExchange Logger for z/OS [16](#)
 - PowerExchange Navigator [16](#)

P

- PC_AUTH statement
 - DBMOVER configuration file [164](#)
- personal certificate, creating [311](#)
- PG CAPI_CONNECTION parameters
 - parameters and syntax [82](#)
- PIPE statement
 - DBMOVER configuration file [165](#)
- POLLTIME statement
 - DBMOVER configuration file [165](#)
- PostgreSQL DBMOVER statements
 - DBMOVER configuration file [42](#)
- PowerExchange
 - components [16](#)
 - configuring operating system authentication [171](#), [251](#)
 - encryption [321](#)
 - ODBC driver [386](#)
 - overview [16](#)
- PowerExchange Agent
 - overview [16](#)
- PowerExchange Condense
 - overview [16](#)
- PowerExchange Data Source tab
 - in PowerExchange Data Source wizard [386](#)
- PowerExchange Data Source wizard
 - AS/400 Properties tab [392](#)
 - CAPX Properties tab [393](#)
 - CAPXRT Properties tab [395](#)
 - DB2/S390 Properties tab [397](#)
 - General tab [389](#)
 - overview [386](#)
 - PowerExchange Data Source tab [386](#)
- PowerExchange datatypes
 - DTLNET_BIN [423](#)
 - DTLNET_BIT [423](#)
 - DTLNET_CHAR [423](#)
 - DTLNET_DATE [423](#)
 - DTLNET_DOUBLE [423](#)
 - DTLNET_FLOAT [423](#)
 - DTLNET_MONEY [423](#)
 - DTLNET_NUM16 [423](#)
 - DTLNET_NUM16U [423](#)
 - DTLNET_NUM32 [423](#)
 - DTLNET_NUM32U [423](#)
 - DTLNET_NUM64 [423](#)
 - DTLNET_NUM64U [423](#)
 - DTLNET_NUM8 [423](#)
 - DTLNET_NUM8U [423](#)
 - DTLNET_NUMCHAR [423](#)
 - DTLNET_PACKED [423](#)
 - DTLNET_STRING [423](#)
 - DTLNET_TIME [423](#)
 - DTLNET_TIMESTAMP [423](#)
 - DTLNET_UPACKED [423](#)
 - DTLNET_UZONED [423](#)
 - DTLNET_VARBIN [423](#)
 - DTLNET_VARCHAR [423](#)
 - DTLNET_ZONED [423](#)
- PowerExchange ECCR
 - overview [16](#)
- PowerExchange jobs
 - data set access requirements [265](#)
- PowerExchange Listener
 - overview [16](#)
- PowerExchange Listener commands
 - authorizing users to run (i5/OS) [258](#)
 - authorizing users to run (Windows) [261](#)

- PowerExchange Listener commands (*continued*)
 - security for (Linux, UNIX, and Windows) [260](#)
- PowerExchange Logger for Linux, UNIX, and Windows
 - overview [16](#)
- PowerExchange Logger for z/OS
 - overview [16](#)
- PowerExchange message logs
 - determining the size of alternative log data sets on z/OS [221](#)
 - primary message log file [219](#)
 - types of message log files [218](#)
- PowerExchange Navigator
 - overview [16](#)
- PowerExchange ODBC driver parameters
 - DB2 for z/OS bulk [413](#)
 - general parameters [402](#)
 - Linux [401](#)
 - UNIX [401](#)
- PowerExchange ODBC drivers [383](#)
- PowerExchange processes
 - configuring to receive pwxcmd commands [206](#)
- PowerExchange security [250](#)
- PowerExchange selective sign-on
 - security [293](#)
- PowerExchange sign-on file
 - security [294](#)
- PRE861_COMPAT statement
 - DBMOVER configuration file [166](#)
- PRGIND statement
 - DBMOVER configuration file [166](#)
- PRGINT statement
 - DBMOVER configuration file [167](#)
- pwxcmd commands
 - authorizing users to run [171](#), [251](#)
 - authorizing users to run (i5/OS) [259](#)
 - authorizing users to run (Linux, UNIX, and Windows) [261](#)
 - authorizing users to run (z/OS) [278](#)
 - configuration [206](#)
 - setting up command-handling service for [186](#)
 - sign-on file examples [299](#)
- pwxcmd program
 - authenticating users for use of (i5/OS) [259](#)
 - authenticating users for use of (z/OS) [277](#)
 - security for [262](#)
 - security for (Linux, UNIX, and Windows) [260](#)
- PWXSOMAXCONN statement
 - DBMOVER configuration file [167](#)

R

- RACF class security
 - for capture registrations [270](#)
 - for extraction maps [270](#)
- RACF_CLASS statement
 - DBMOVER configuration file [168](#)
- RDBMSINSRTDFLT statement
 - DBMOVER configuration file [168](#)
- RECFM statement
 - DBMOVER configuration file [169](#)
- REJECT_FILE_DELIMITER statement
 - DBMOVER configuration file [169](#)
- RELEASE statement
 - DBMOVER configuration file [170](#)
- resource access requirements
 - i5/OS [256](#)
- resource profiles
 - capture registrations [270](#)
 - extraction maps [270](#)

RMTRDBDIRE statement
 DBMOVER configuration file [170](#)
RMTSYSNAME statement
 DBMOVER configuration file [170](#)

S

security

 Adabas [274](#)
 capture registrations on z/OS [269](#)
 configuring PowerExchange selective sign-on [293](#)
 data maps [273](#)
 Datacom table-level [275](#)
 Datacom, table-level [275](#)
 DB2 for z/OS [276](#)
 example for capture registrations on z/OS [272](#)
 example for extraction maps on z/OS [272](#)
 extraction maps on z/OS [269](#)
 i5/OS [255](#), [256](#)
 IMS [277](#)
 infacmd pwx program [262](#)
 infacmd pwx program (Linux, UNIX, and Windows) [260](#)
 Linux [260](#)
 PowerExchange [250](#)
 PowerExchange Listener commands (Linux, UNIX, and Windows) [260](#)
 PowerExchange passphrases [279](#)
 PowerExchange selective sign-on [293](#)
 PowerExchange sign-on file [294](#)
 pwxcmd program [259](#), [262](#), [277](#)
 pwxcmd program (Linux, UNIX, and Windows) [260](#)
 selective sign-on (Linux, UNIX, and Windows) [260](#)
 UNIX [260](#)
 Windows [260](#)
 z/OS [263](#)

SECURITY statement
 DBMOVER configuration file [171](#), [251](#), [269](#)

SECURITY_MSGSUPP statement
 DBMOVER configuration file [175](#)

selective sign-on
 configuring [293](#)
 security [293](#)

SERVICE_TIMEOUT statement
 DBMOVER configuration file [175](#)

SESSID statement
 DBMOVER configuration file [176](#)

SHOW_THREAD_PERF statement
 DBMOVER configuration file [176](#)

sign-on file
 AUTHGROUP statement [295](#)
 AUTHGROUP statements in [295](#)
 examples [299](#)
 security [294](#)
 syntax for statements in [294](#)
 USER statement [297](#)
 USER statements in [295](#)

SMF
 record format [240](#)
 standard SMF header with sub type included [240](#)
 statistics logging and reporting [227](#)

SMF record format
 extended section descriptor [242](#)
 extended section for data access [246](#)
 extended section for PowerExchange Client [245](#)
 extended section for PowerExchange Listener [244](#)
 function/access method section (DB2) [246](#)
 general section [243](#)

SMF record format (*continued*)
 PowerExchange header/descriptor [242](#)
 SMF triplet section descriptor [241](#)

SMF statistics
 Defining the SMF data sets to use for reporting [231](#)
 example reports [232](#)
 generating reports from SMF archived data sets [230](#)
 JCL for generating reports [231](#)
 logging and reporting statistics [227](#)
 viewing statistics written to a file [239](#)
 viewing statistics written to SMF [235](#)
 writing statistics to SMF on z/OS [228](#)
 writing to a data set on z/OS [229](#)

source-specific CAPI_CONNECTION statements
 DBMOVER configuration file [205](#)

SPACE statement
 DBMOVER configuration file [177](#)

SQL escape sequences
 DTLARRAYSIZE [416](#)
 DTLDSN [417](#)
 DTLEVENTTB [418](#)
 DTLFILELIST [418](#)
 DTLIMGOV [418](#)
 DTLIMTYPE [418](#)
 DTLINSMODE [418](#)
 DTLJRNL [418](#)
 DTLLIBRARYLIST [419](#)
 DTLLOGSID [419](#)
 DTLMODELCB [419](#)
 DTLNOUPDATECEP [419](#)
 DTLORACOLL [419](#)
 DTLORACONN [419](#)
 DTLORAINST [419](#)
 DTLORASchema [420](#)
 DTLOVRDBF [420](#)
 DTLREJECTFILE [420](#)
 DTLSESSID [420](#)
 DTLSTOPONERRORS [420](#)
 DTLSTRIPORDERBY [420](#)
 DTLTIMEOUT [421](#)
 DTLUDBDB [421](#)
 DTLUPDELSEINS [421](#)
 DTLXTRASchema [421](#)
 DTLXTYPE [421](#)
 ODBC [414](#), [415](#)
 with PowerCenter [415](#)

SSL
 architecture, PowerExchange [301](#)
 security on PowerExchange network [300](#)
 support [300](#)

SSL statement
 DBMOVER configuration file [177](#)
SSL_ALLOW_SELF_SIGNED statement
 DBMOVER configuration file [178](#)
SSL_CIPHER_LIST statement
 DBMOVER configuration file [179](#)
SSL_CONTEXT_METHOD statement
 DBMOVER configuration file [180](#)
SSL_REQ_CLNT_CERT statement
 DBMOVER configuration file [180](#)
SSL_REQ_SRVR_CERT statement
 DBMOVER configuration file [181](#)
START_UP_USER_EXIT statement
 DBMOVER configuration file [181](#)
STATS statement
 DBMOVER configuration file [182](#)
SUBMITTIMEOUT statement
 DBMOVER configuration file [184](#)

summary

DBMOVER statements [19](#)

SUP_FUNC statement

DBMOVER configuration file [184](#)

SUP_REQUEST statement

DBMOVER configuration file [185](#)

SUP_SSNAME statement

DBMOVER configuration file [185](#)

SUP_SSTYPE statement

DBMOVER configuration file [185](#)

SUPPRESS_DATA_LOGGING statement

DBMOVER configuration file [186](#)

SVCNODE statement

DBMOVER configuration file [186](#), [206](#)

syntax

PowerExchange sign-on file [294](#)

syntax rules

DBMOVER statements [49](#)

SYSOUT_TIMESTAMP statement

DBMOVER configuration file [188](#)

T

TAPEWAIT statement

DBMOVER configuration file [188](#)

TCPIP_ACCEPT_NONBLOCK statement

DBMOVER configuration file [188](#)

TCPIP_DIAGNOSTICS_TRACE_SZ statement

DBMOVER configuration file [189](#)

TCPIP_DIAGNOSTICS_TRACE statement

DBMOVER configuration file [189](#)

TCPIP_SHOW_POOLING statement

DBMOVER configuration file [190](#)

TCPIPBUFSIZE statement

DBMOVER configuration file [191](#)

TCPIPVER statement

DBMOVER configuration file [191](#)

TEMPHLQ statement

DBMOVER configuration file [191](#)

TEXT_EOF_FOR_BINARY statement

DBMOVER configuration file [192](#)

TIMEZONE statement

DBMOVER configuration file [192](#)

TRACE statement

DBMOVER configuration file [193](#)

TRACING statement

DBMOVER configuration file [193](#)

U

UDB CAPI_CONNECTION parameters

parameters and syntax [85](#)

UNIT statement

DBMOVER configuration file [199](#)

UNIX

security [260](#)

UOWC CAPI_CONNECTION parameters

parameters and syntax [89](#)

USE_DB_AUTH statement

DBMOVER configuration file [199](#)

USE_TYPE1_FALLBACKS statement

DBMOVER configuration file [200](#)

user credentials

authenticating (Linux and UNIX) [262](#)

authenticating (Windows) [262](#)

USER statement

sign-on file [295](#), [297](#)

USESUP statement

DBMOVER configuration file [200](#)

V

VOLSER statement

DBMOVER configuration file [201](#)

VSAM DBMOVER statements

DBMOVER configuration file [42](#)

VSAM statement

DBMOVER configuration file [201](#)

W

WAITDSN statement

DBMOVER configuration file [202](#)

Windows

security [260](#)

WRT_ERROR_HANDLING statement

DBMOVER configuration file [203](#)

Z

z/OS

security [263](#)

z/OS DBMOVER statements

DBMOVER configuration file [46](#)