Version 9.6.1
August 2014

Copyright (c) 1998-2014 Informatica Corporation. All rights reserved.

This software and documentation contain proprietary information of Informatica Corporation and are provided under a license agreement containing restrictions on use and disclosure and are also protected by copyright law. Reverse engineering of the software is prohibited. 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 Corporation. This Software may be protected by U.S. and/or international Patents and other Patents Pending.

Use, duplication, or disclosure of the Software by the U.S. Government is subject to the restrictions set forth in the applicable software license agreement and as provided in DFARS 227.7202-1(a) and 227.7702-3(a) (1995), DFARS 252.227-7013(c)(1)(ii) (OCT 1988), FAR 12.212(a) (1995), FAR 52.227-19, or FAR 52.227-14 (ALT III), as applicable.

The information in this product or documentation is subject to change without notice. If you find any problems in this product or documentation, please report them to us in writing.

Informatica, Informatica Platform, Informatica Data Services, PowerCenter, PowerCenterRT, PowerCenter Connect, PowerCenter Data Analyzer, PowerExchange, PowerMart, Metadata Manager, Informatica Data Quality, Informatica Data Explorer, Informatica B2B Data Transformation, Informatica B2B Data Exchange Informatica On Demand, Informatica Identity Resolution, Informatica Application Information Lifecycle Management, Informatica Complex Event Processing, Ultra Messaging and Informatica Master Data Management are trademarks or registered trademarks of Informatica Corporation in the United States and in jurisdictions throughout the world. All 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, including without limitation: Copyright DataDirect Technologies. All rights reserved. Copyright © Sun Microsystems. All rights reserved. Copyright © RSA Security Inc. All Rights Reserved. Copyright © Ordinal Technology Corp. All rights reserved. Copyright © Aandacht c.v. All rights reserved. Copyright Genivia, Inc. All rights reserved. Copyright Isomorphic Software. All rights reserved. Copyright © Meta Integration Technology, Inc. All rights reserved. Copyright © Intalio. All rights reserved. Copyright © Oracle. All rights reserved. Copyright © Adobe Systems Incorporated. All rights reserved. Copyright © DataArt, Inc. All rights reserved. Copyright © ComponentSource. All rights reserved. Copyright © Microsoft Corporation. All rights reserved. Copyright © Rogue Wave Software, Inc. All rights reserved. Copyright © Teradata Corporation. All rights reserved. Copyright © Yahoo Inc. All rights reserved. Copyright © Glyph & Cog, LLC. All rights reserved. Copyright © Thinkmap, Inc. All rights reserved. Copyright © Clearspace Limited. All rights reserved. Copyright © Information Builders, Inc. All rights reserved. Copyright © OSS Nakolva, Inc. All rights reserved. Copyright © Edifecs, Inc. All rights reserved. Copyright © Olexo Communications, Inc. All rights reserved. Copyright © International Organization for Standardization 1986. All rights reserved. Copyright © ej-technologies GmbH. All rights reserved. Copyright © Jaspersoft Corporation. All rights reserved. Copyright © International Business Machines Corporation. All rights reserved. Copyright © EMC Corporation. All rights reserved. Copyright © Flexera Software. All rights reserved. Copyright © Jinfonet Software. All rights reserved. Copyright © Apple Inc. All rights reserved. Copyright © Telek In. All rights reserved. Copyright © BEA Systems. All rights reserved. Copyright © PDFlib GmbH. All rights reserved. Copyright © JBoss. All rights reserved. Copyright © Sencha, Inc. All rights reserved.

This product includes software developed by the Apache Software Foundation (http://www.apache.org/), and/or other software which is licensed under various versions of the Apache License (the "License"). You may obtain a copy of these Licenses at http://www.apache.org/licenses/. Unless required by applicable law or agreed to in writing, software distributed under these Licenses is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

This product includes software which was developed by Mozilla (http://www.mozilla.org/), software copyright The JBoss Group, LLC, all rights reserved; software copyright © 1999-2006 by Bruno Lowagie and Paulo Soares and other software which is licensed under various versions of the GNU Lesser General Public License Agreements, which may be found at http://www.gnu.org/licenses/lgpl.html. The materials are provided free of charge by Informatica, "as-is", without warranty of any kind, either express or implied, including but not limited to the implied warranties of merchantability and fitness for a particular purpose.

The product includes ACE(TM) and TAO(TM) software copyrighted by Douglas C. Schmidt and his research group at Washington University, University of California, Irvine, and Vanderbilt University, Copyright © 1993-2006, all rights reserved.

This product includes software developed by the OpenSSL Project for use in the OpenSSL Toolkit (copyright The OpenSSL Project. All Rights Reserved) and redistribution of this software is subject to terms available at http://www.openssl.org and http://www.openssl.org/source/license.html.

This product includes Curl software which is Copyright © 1996-2013, Daniel Stenberg, <daniel@haxx.se>. All Rights Reserved. Permissions and limitations regarding this software are subject to terms available at http://curl.haxx.se/docs/copyright.html. Permission to use, copy, modify, and distribute this software for any purpose with or without fee is hereby granted, provided that the above copyright notice and this permission notice appear in all copies.

This product includes software copyright © 2001-2005 (©) MetaStuff, Ltd. All Rights Reserved. Permissions and limitations regarding this software are subject to terms available at http://www.dom4j.org/license.html.

The product includes software copyright © 2004-2007, The Dojo Foundation. All Rights Reserved. Permissions and limitations regarding this software are subject to terms available at http://dojotoolkit.org/license.

This product includes ICU software which is copyright International Business Machines Corporation and others. All rights reserved. Permissions and limitations regarding this software are subject to terms available at http://source.icu-project.org/repos/icu/icu/trunk/license.html.

This product includes software copyright © 1996-2006 Per Bothner. All rights reserved. Your right to use such materials is set forth in the license which may be found at http://www.gnu.org/software/kawa/Software-License.html.

This product includes OSSP UUID software which is Copyright © 2002 Ralf S. Engelschall, Copyright © 2002 The OSSP Project Copyright © 2002 Cable & Wireless Deutschland. Permissions and limitations regarding this software are subject to terms available at http://www.opensource.org/licenses/mt-license.php.

This product includes software developed by Boost (http://www.boost.org/) or under the Boost software license. Permissions and limitations regarding this software are subject to terms available at http://www.boost.org/LICENSE_1_0.txt.

This product includes software copyright © 1997-2007 University of Cambridge. Permissions and limitations regarding this software are subject to terms available at http://www.pcre.org/license.txt.


This product includes software licensed under the Academic Free License (http://www.opensource.org/licenses/afl-3.0.php), the Common Development and Distribution License (http://www.opensource.org/licenses/cddl1.php) the Common Public License (http://www.opensource.org/licenses/cpl1.0.php), the Sun Binary Code License Agreement Supplemental License Terms, the BSD License (http://www.opensource.org/licenses/bsd-license.php), the new BSD License (http://opensource.org/licenses/BSD-3-Clause), the MIT License (http://www.opensource.org/licenses/mit-license.php), the Artistic License (http://www.opensource.org/licenses/artistic-license-1.0) and the Initial Developer's Public License Version 1.0 (http://www.firebirdsql.org/en/initial-developer-s-public-license-version-1-0/).

This product includes software copyright © 2003-2006 Joe Walnes, 2006-2007 XStream Committers. All rights reserved. Permissions and limitations regarding this software are subject to terms available at http://xstream.codehaus.org/license.html. This product includes software developed by the Indiana University Extreme! Lab. For further information please visit http://www.extreme.indiana.edu/.

This product includes software Copyright (c) 2013 Frank Balluffi and Markus Moeller. All rights reserved. Permissions and limitations regarding this software are subject to terms of the MIT license.

This Software is protected by U.S. Patent Numbers 5,794,246; 6,014,670; 6,016,501; 6,029,178; 6,032,158; 6,035,307; 6,044,374; 6,092,086; 6,208,990; 6,339,775; 6,460,226; 6,789,096; 6,823,373; 6,850,947; 6,895,471; 7,117,215; 7,162,643; 7,243,110; 7,254,590; 7,281,001; 7,421,458; 7,496,888; 7,523,121; 7,584,422; 7,676,516; 7,720,842; 7,721,270; 7,774,791; 8,065,296; 8,150,803; 8,166,048; 8,166,071; 8,253,142; 8,260,622; 8,264,973; 8,271,477; 8,327,419; 8,386,435; 8,392,440; 8,453,159; 8,458,230; and RE44,478, International Patents and other Patents Pending.

DISCLAIMER: Informatica Corporation provides this documentation "as is" without warranty of any kind, either express or implied, including, but not limited to, the implied warranties of noninfringement, merchantability, or use for a particular purpose. Informatica Corporation does not warrant that this software or documentation is error free. The information provided in this software or documentation may include technical inaccuracies or typographical errors. The information in this software and documentation is subject to change at any time without notice.

NOTICES

This Informatica product (the "Software") includes certain drivers (the "DataDirect Drivers") from DataDirect Technologies, an operating company of Progress Software Corporation ("DataDirect") which are subject to the following terms and conditions:

1. THE DATADIRECT DRIVERS ARE PROVIDED "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT.

2. IN NO EVENT WILL DATADIRECT OR ITS THIRD PARTY SUPPLIERS BE LIABLE TO THE END-USER CUSTOMER FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, CONSEQUENTIAL OR OTHER DAMAGES ARISING OUT OF THE USE OF THE ODBC DRIVERS, WHETHER OR NOT INFORMED OF THE POSSIBILITIES OF DAMAGES IN ADVANCE. THESE LIMITATIONS APPLY TO ALL CAUSES OF ACTION, INCLUDING, WITHOUT LIMITATION, BREACH OF CONTRACT, BREACH OF WARRANTY, NEGLIGENCE, STRICT LIABILITY, MISREPRESENTATION AND OTHER TORTS.

Part Number: PWX-REF-961-0001
# Table of Contents

**Preface** ................................................................. xi

Informatica Resources .................................................... xi

Informatica My Support Portal ........................................ xi

Informatica Documentation ............................................. xii

Informatica Web Site .................................................. xii

Informatica How-To Library ............................................. xii

Informatica Knowledge Base ......................................... xii

Informatica Support YouTube Channel ................................ xii

Informatica Marketplace ............................................... xii

Informatica Velocity .................................................. xiii

Informatica Global Customer Support ............................... xiii

**Chapter 1: Introduction to PowerExchange** ....................... 1

PowerExchange Overview .............................................. 1

PowerExchange Components .......................................... 1

**Chapter 2: DBMOVER Configuration File** ....................... 4

DBMOVER Configuration File Overview ............................ 4

DBMOVER Statement Summary Tables ................................ 4

Summary of All DBMOVER Statements ................................ 4

Summary of Data Source-Specific DBMOVER Statements ........... 22

Summary of Operating System-Specific DBMOVER Statements .... 28

Syntax of DBMOVER Statements ...................................... 33

Syntax Rules and Guidelines for DBMOVER Statements .......... 33

Notational Conventions for DBMOVER Statements ................ 34

DBMOVER Statements .................................................. 34

ABEND_SW Statement .................................................. 34

ADA_L3_ALLOW Statement .............................................. 35

ADABAS_DEFAULT_DBID Statement .................................. 36

ADABAS_PREFIX Statement ............................................ 36

ADABASCODEPAGE Statement ......................................... 37

ADAOPT Statement .................................................... 38

ADAOPTM Statement ................................................... 39

ADAPREFETCH Statement ............................................. 39

ADASTATS Statement ................................................ 39

ADAUSER Statement ................................................... 40

APPBUFSIZE Statement ............................................... 40

APPBUFSIZEEDYN Statement ......................................... 41

AS400EVENTMSGQ Statement ........................................ 41

AS400USRJRNCODE Statement ....................................... 42
<table>
<thead>
<tr>
<th>Statement</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>BS Statement</td>
<td>43</td>
</tr>
<tr>
<td>CAPI_CONN_NAME Statement</td>
<td>43</td>
</tr>
<tr>
<td>CAPI_CONNECTION - AS4J Statement</td>
<td>44</td>
</tr>
<tr>
<td>CAPI_CONNECTION - CAPX Statement</td>
<td>49</td>
</tr>
<tr>
<td>CAPI_CONNECTION - LRAP Statement</td>
<td>51</td>
</tr>
<tr>
<td>CAPI_CONNECTION - MSQL Statement</td>
<td>53</td>
</tr>
<tr>
<td>CAPI_CONNECTION - ORAD Statement</td>
<td>56</td>
</tr>
<tr>
<td>CAPI_CONNECTION - ORCL Statement</td>
<td>58</td>
</tr>
<tr>
<td>CAPI_CONNECTION - UDB Statement</td>
<td>64</td>
</tr>
<tr>
<td>CAPI_CONNECTION - UOWC Statement</td>
<td>68</td>
</tr>
<tr>
<td>CAPI_SRC_DFLT Statement</td>
<td>73</td>
</tr>
<tr>
<td>CAPT_PATH Statement</td>
<td>74</td>
</tr>
<tr>
<td>CAPT_XTRA Statement</td>
<td>75</td>
</tr>
<tr>
<td>CMDNODE Statement</td>
<td>75</td>
</tr>
<tr>
<td>CODEPAGE Statement</td>
<td>76</td>
</tr>
<tr>
<td>COLON Statement</td>
<td>77</td>
</tr>
<tr>
<td>COMPRESS Statement</td>
<td>78</td>
</tr>
<tr>
<td>CONSOLE_CODEPAGE Statement</td>
<td>78</td>
</tr>
<tr>
<td>CONSOLE_TRACE Statement</td>
<td>78</td>
</tr>
<tr>
<td>CONVCHAR Statement</td>
<td>79</td>
</tr>
<tr>
<td>CPX_DIR Statement</td>
<td>80</td>
</tr>
<tr>
<td>CREDENTIALS_CASE Statement</td>
<td>80</td>
</tr>
<tr>
<td>DATAMAP_SERVER Statement</td>
<td>81</td>
</tr>
<tr>
<td>DATERANGE Statement</td>
<td>82</td>
</tr>
<tr>
<td>DB2_BIN_AS_CHAR Statement</td>
<td>82</td>
</tr>
<tr>
<td>DB2_BIN_CODEPAGE Statement</td>
<td>83</td>
</tr>
<tr>
<td>DB2_ERRORFILE Statement</td>
<td>83</td>
</tr>
<tr>
<td>DB2CODEPAGE Statement</td>
<td>84</td>
</tr>
<tr>
<td>DB2DEF_ENCODING Statement</td>
<td>88</td>
</tr>
<tr>
<td>DB2ID Statement</td>
<td>89</td>
</tr>
<tr>
<td>DB2PLAN Statement</td>
<td>89</td>
</tr>
<tr>
<td>DECPPOINT Statement</td>
<td>90</td>
</tr>
<tr>
<td>DEFAULTCHAR Statement</td>
<td>90</td>
</tr>
<tr>
<td>DEFAULTDATE Statement</td>
<td>91</td>
</tr>
<tr>
<td>DISABLE_PARTITIONS Statement</td>
<td>91</td>
</tr>
<tr>
<td>DISP Statement</td>
<td>91</td>
</tr>
<tr>
<td>DM_RESOURCE Statement</td>
<td>92</td>
</tr>
<tr>
<td>DM_SUBTASK Statement</td>
<td>93</td>
</tr>
<tr>
<td>DMXCACHE_DELETEECSA Statement</td>
<td>94</td>
</tr>
<tr>
<td>DMXCACHE_MAX_MEMORY_MB Statement</td>
<td>94</td>
</tr>
<tr>
<td>DMXCACHE_MULTIPLEJOBS Statement</td>
<td>95</td>
</tr>
<tr>
<td>DMX_DIR Statement</td>
<td>96</td>
</tr>
<tr>
<td>Statement</td>
<td>Page</td>
</tr>
<tr>
<td>------------------------------------------------</td>
<td>------</td>
</tr>
<tr>
<td>DTLMSG_CODEPAGE Statement.</td>
<td>96</td>
</tr>
<tr>
<td>ENABLE_AUTO_COMMIT Statement.</td>
<td>97</td>
</tr>
<tr>
<td>ENCRYPT Statement.</td>
<td>97</td>
</tr>
<tr>
<td>ENCRYPTLEVEL Statement.</td>
<td>98</td>
</tr>
<tr>
<td>ENQMAJORNAME Statement.</td>
<td>99</td>
</tr>
<tr>
<td>ENQSYSTEMS Statement.</td>
<td>99</td>
</tr>
<tr>
<td>ERRROWNOTFOUND Statement.</td>
<td>100</td>
</tr>
<tr>
<td>EXT_CP_SUPPT Statement.</td>
<td>100</td>
</tr>
<tr>
<td>GDGLOCATE Statement.</td>
<td>101</td>
</tr>
<tr>
<td>GSBUFSIZE Statement.</td>
<td>102</td>
</tr>
<tr>
<td>ICUALIAS Statement.</td>
<td>102</td>
</tr>
<tr>
<td>ICUCNVPROPERTY Statement.</td>
<td>103</td>
</tr>
<tr>
<td>ICUCONVERTER Statement.</td>
<td>105</td>
</tr>
<tr>
<td>ICUDATADIR Statement.</td>
<td>107</td>
</tr>
<tr>
<td>IMSID Statement.</td>
<td>107</td>
</tr>
<tr>
<td>JOBCLASS Statement.</td>
<td>108</td>
</tr>
<tr>
<td>LISTENER Statement.</td>
<td>109</td>
</tr>
<tr>
<td>LOADCTLFILE Statement.</td>
<td>110</td>
</tr>
<tr>
<td>LOADJOBFILE Statement.</td>
<td>111</td>
</tr>
<tr>
<td>LOG_CODEPAGE Statement.</td>
<td>111</td>
</tr>
<tr>
<td>LOG_LINE_LIMIT Statement.</td>
<td>112</td>
</tr>
<tr>
<td>LOGPATH Statement.</td>
<td>112</td>
</tr>
<tr>
<td>LOGSID Statement.</td>
<td>113</td>
</tr>
<tr>
<td>LOWVALUES Statement.</td>
<td>113</td>
</tr>
<tr>
<td>LRECL Statement.</td>
<td>114</td>
</tr>
<tr>
<td>MAXTASKS Statement.</td>
<td>114</td>
</tr>
<tr>
<td>MSGPREFIX Statement.</td>
<td>115</td>
</tr>
<tr>
<td>MSGPREFIX-HYPHEN Statement.</td>
<td>115</td>
</tr>
<tr>
<td>MSS_ERRORFILE Statement.</td>
<td>116</td>
</tr>
<tr>
<td>MVSDB2AF Statement.</td>
<td>116</td>
</tr>
<tr>
<td>NEGSIGN Statement.</td>
<td>117</td>
</tr>
<tr>
<td>NETPORT Statement.</td>
<td>117</td>
</tr>
<tr>
<td>NODE Statement.</td>
<td>120</td>
</tr>
<tr>
<td>NOGETHOSTBYNAME Statement.</td>
<td>121</td>
</tr>
<tr>
<td>NRDB_WRITE_CHAR_NULL_FILL Statement.</td>
<td>122</td>
</tr>
<tr>
<td>NRDB_WRITE_NUM_NULL_FILL Statement.</td>
<td>122</td>
</tr>
<tr>
<td>NUMERICSIGN Statement.</td>
<td>123</td>
</tr>
<tr>
<td>ODBASUPP Statement.</td>
<td>123</td>
</tr>
<tr>
<td>ORA_ERRORFILE Statement.</td>
<td>123</td>
</tr>
<tr>
<td>ORACLE_CAPTURE_TYPE Statement.</td>
<td>124</td>
</tr>
<tr>
<td>ORACLE_UNHANDLED_NUMASCHAR Statement.</td>
<td>125</td>
</tr>
<tr>
<td>ORACLECODEPAGE Statement.</td>
<td>126</td>
</tr>
</tbody>
</table>
ORACLEID Statement. .......................................................... 127
OUSP Statement. ............................................................... 129
PC_AUTH Statement. ............................................................ 130
PIPE Statement. ............................................................. 130
POLLTIME Statement. .......................................................... 130
PRGIND Statement. ............................................................ 131
PRGINT Statement. ............................................................. 131
PWXSMAXCONN Statement. .................................................. 131
RACF_CLASS Statement. ...................................................... 132
RDBMSINSRTDFLT Statement. ............................................. 132
RECFM Statement. ............................................................. 133
REJECT_FILE_DELIMITER Statement. .................................... 133
RELEASE Statement. .......................................................... 134
RMTRDBDIRE Statement. ..................................................... 134
RMTSYSNAME Statement. .................................................... 134
SECURITY Statement. .......................................................... 135
SERVICE_TIMEOUT Statement. .......................................... 138
SESSID Statement. ............................................................ 138
SHOW_THREAD_PERF Statement. ......................................... 139
SPACE Statement. ............................................................. 139
SSL Statement. ................................................................. 140
SSL_ALLOW_SELFSIGNED Statement. .................................... 141
SSL_CIPHER_LIST Statement. .............................................. 141
SSL_CONTEXT_METHOD Statement. ....................................... 142
SSL_REQ_CLNT_CERT Statement. ......................................... 143
SSL_REQ_SRVR_CERT Statement. ......................................... 143
STATS Statement. .............................................................. 144
SUBMITTIMEOUT Statement. ................................................ 145
SUP_FUNC Statement. .......................................................... 146
SUP_REQUEST Statement. .................................................... 146
SUP_SSNAME Statement. ...................................................... 146
SUP_SSTYPE Statement. ...................................................... 147
SUPPRESS_DATA_LOGGING Statement. .................................. 147
SVCNODE Statement. .......................................................... 147
SYSOUT_TIMESTAMP Statement. .......................................... 149
TAPEWAIT Statement. .......................................................... 149
TCPIP_DIAGNOSTICS_TRACE Statement. ................................ 149
TCPIP_DIAGNOSTICS_TRACE_SZ Statement. .......................... 150
TCPIP_SHOW_POOLING Statement. ......................................... 150
TCPIPBUFSIZE Statement. .................................................... 151
TCPIPVER Statement. .......................................................... 151
TEMPHLPQ Statement. .......................................................... 151
Chapter 7: Secure Sockets Layer Support .............................................. 231
SSL Security on a PowerExchange Network. ........................................ 231
PowerExchange SSL Architecture. ....................................................... 232
FIPS 140-2 Compliance. ....................................................................... 232
  FIPS 140-2 Compliant Cipher Suites. ................................................ 233
  FIPS 140-2 Compliance Considerations on z/OS. ................................. 234
PWXUSSL Utility. .................................................................................. 234
PowerExchange SSL Configuration Steps. ............................................ 234
SSL Server Configuration on z/OS. ....................................................... 235
  Updating the AT-TLS Policy File. ....................................................... 235
  Creating a Personal Certificate on z/OS. ............................................. 238
  Configuring the PowerExchange Listener on the z/OS SSL Server. ......... 239
SSL Client and Server Configuration on Linux, UNIX, and Windows. ............ 239
  OpenSSL in a PowerExchange SSL Environment. ............................... 240
  Creating a CA Certificate with OpenSSL. ........................................... 240
  Creating a Personal Certificate with OpenSSL. ................................... 241
  Customizing the DBMOVER Configuration File on the SSL Server. ....... 242
  Customizing the DBMOVER Configuration File on the SSL Client. ......... 243
Verifying Connections. ........................................................................... 244

Chapter 8: PowerExchange Alternative Network Security ......................... 245
PowerExchange Alternative Network Security Overview. .......................... 245
Modes of PowerExchange Encryption. .................................................. 245
  PowerExchange Encryption Standard. ................................................ 246
  DES Encryption Standard. ................................................................. 246
  Triple DES Encryption Standard. ....................................................... 246
  RC2 Cipher. ........................................................................................ 246
Setting Encryption Defaults in the DBMOVER Configuration File. ................. 247

Chapter 9: PowerExchange Nonrelational SQL ........................................ 248
PowerExchange Nonrelational SQL Overview. ........................................ 248
PowerExchange Nonrelational SQL Syntax. ............................................ 248
  SQL Syntax for a Basic Comparison. ................................................... 249
  LIKE Comparison. ............................................................................. 250
  BETWEEN Comparison. .................................................................... 251
  NULL Comparison. ............................................................................ 251
  Allowed Keywords in the SQL Syntax. ................................................. 251
  Accessing Data with Column Names the Same as SQL Keywords. .......... 251
  Using SQL with User-Defined Fields. ............................................... 251
  IMS Call Considerations. .................................................................... 251
DTLDESCRIBE Metadata Syntax. ............................................................. 252
  DTLDESCRIBE TABLES Qualifier. ....................................................... 252
Chapter 11: Using the PowerExchange ODBC Drivers

PowerExchange ODBC Driver Overview

PowerExchange ODBC Data Sources on Windows

Creating an ODBC Data Source on Windows

Accessing Multibyte Metadata with the PowerExchange ODBC Unicode Driver on Windows

PowerExchange Data Source Wizard Properties

PowerExchange ODBC Data Sources on Linux or UNIX

Example PowerExchange Data Source Specifications

Accessing Multibyte Metadata with ODBC

PowerExchange ODBC Driver Data Source Parameters

SQL Escape Sequences

SQL Escape Sequences with PowerCenter

SQL Escape Sequences Available in PowerCenter

{DTLADAPREFETCH=Y[N]} Escape Sequence

{DTLADAPWD=} Escape Sequence

{DTLAPP=} Escape Sequence

{DTLCONFWRITE=N|Y[T]} Escape Sequence

{DTLCONNOVR=} Escape Sequence

{DTLDATAPWD=} Escape Sequence

{DTLDB2DEGREE=} Escape Sequence

{DTLDSPN=} Escape Sequence

{DTLEVENTTB=} Escape Sequence

{DTLIMTYPE=BA|AI|TU} Escape Sequence

{DTLIMGOV=Y|N} Escape Sequence

{DTLINSMODE=LOAD|UPDATE} Escape Sequence

{DTLJRNL=} Escape Sequence

{DTLIBRARYLIST=} Escape Sequence

{DTLNOUPDATECDEP=Y|N} Escape Sequence

{DTLORACOLL=} Escape Sequence

{DTLORACONN=} Escape Sequence

{DTLORAINST=} Escape Sequence

{DTLORASHEMA=} Escape Sequence

{DTLOVRDBF=} Escape Sequence

{DTLREJECTFILE=} Escape Sequence

{DTLSESSID=} Escape Sequence

{DTLSTOPONERRORS=} Escape Sequence
Chapter 12: PowerExchange Datatypes and Conversion Matrix.............. 346
PowerExchange Datatype Conversions. ........................................ 346
ODBC and PowerExchange Datatype Equivalencies. ....................... 347

Appendix A: DTL__CAPXTIMESTAMP Time Stamps....................... 349
Time Stamps That Are Reported in the DTL__CAPXTIMESTAMP Field by Data Source. ..... 349

Appendix B: PowerExchange Glossary................................. 351

Index. .................................................................................. 370
Preface


The manual includes information about configuring the DBMOVER configuration file, PowerExchange security, and the PowerExchange ODBC driver. The manual also contains information that you need to use PowerExchange in certain environments. For example, if you have nonrelational data sources, the manual describes the SQL that PowerExchange supports to access that data.

Use this manual in conjunction with the PowerExchange Installation and Upgrade Guide, PowerExchange Bulk Data Movement Guide, and PowerExchange CDC guides to gain a complete understanding of the product.

This manual pertains to the following operating systems and databases that PowerExchange supports:

- DB2® for i5/OS® and flat files on i5/OS
- Linux, UNIX, and Windows data sources:
  - DB2 for Linux®, UNIX®, and Windows®
  - Flat files on Linux, UNIX, and Windows
  - Microsoft SQL Server®
  - Oracle®
- z/OS data sources:
  - Adabas®
  - CA Datacom®
  - CA IDMS™
  - DB2 for z/OS®
  - IMS™
  - VSAM and sequential data sets

Informatica Resources

Informatica My Support Portal

The site contains product information, user group information, newsletters, access to the Informatica customer support case management system (ATLAS), the Informatica How-To Library, the Informatica Knowledge Base, Informatica Product Documentation, and access to the Informatica user community.

Informatica Documentation

The Informatica Documentation team takes every effort to create accurate, usable documentation. If you have questions, comments, or ideas about this documentation, contact the Informatica Documentation team through email at infa_documentation@informatica.com. We will use your feedback to improve our documentation. Let us know if we can contact you regarding your comments.

The Documentation team updates documentation as needed. To get the latest documentation for your product, navigate to Product Documentation from http://mysupport.informatica.com.

Informatica Web Site

You can access the Informatica corporate web site at http://www.informatica.com. The site contains information about Informatica, its background, upcoming events, and sales offices. You will also find product and partner information. The services area of the site includes important information about technical support, training and education, and implementation services.

Informatica How-To Library

As an Informatica customer, you can access the Informatica How-To Library at http://mysupport.informatica.com. The How-To Library is a collection of resources to help you learn more about Informatica products and features. It includes articles and interactive demonstrations that provide solutions to common problems, compare features and behaviors, and guide you through performing specific real-world tasks.

Informatica Knowledge Base

As an Informatica customer, you can access the Informatica Knowledge Base at http://mysupport.informatica.com. Use the Knowledge Base to search for documented solutions to known technical issues about Informatica products. You can also find answers to frequently asked questions, technical white papers, and technical tips. If you have questions, comments, or ideas about the Knowledge Base, contact the Informatica Knowledge Base team through email at KB_Feedback@informatica.com.

Informatica Support YouTube Channel

You can access the Informatica Support YouTube channel at http://www.youtube.com/user/INFASupport. The Informatica Support YouTube channel includes videos about solutions that guide you through performing specific tasks. If you have questions, comments, or ideas about the Informatica Support YouTube channel, contact the Support YouTube team through email at supportvideos@informatica.com or send a tweet to @INFASupport.

Informatica Marketplace

The Informatica Marketplace is a forum where developers and partners can share solutions that augment, extend, or enhance data integration implementations. By leveraging any of the hundreds of solutions available on the Marketplace, you can improve your productivity and speed up time to implementation on your projects. You can access Informatica Marketplace at http://www.informaticamarketplace.com.
Informatica Velocity

You can access Informatica Velocity at http://mysupport.informatica.com. Developed from the real-world experience of hundreds of data management projects, Informatica Velocity represents the collective knowledge of our consultants who have worked with organizations from around the world to plan, develop, deploy, and maintain successful data management solutions. If you have questions, comments, or ideas about Informatica Velocity, contact Informatica Professional Services at ips@informatica.com.

Informatica Global Customer Support

You can contact a Customer Support Center by telephone or through the Online Support.

Online Support requires a user name and password. You can request a user name and password at http://mysupport.informatica.com.

Chapter 1

Introduction to PowerExchange

This chapter includes the following topics:

- PowerExchange Overview, 1
- PowerExchange Components, 1

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 enable you move bulk data and change data to a variety of data targets.

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 MVS

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.
This chapter includes the following topics:

- **DBMOVER Configuration File Overview**, 4
- **DBMOVER Statement Summary Tables**, 4
- **Syntax of DBMOVER Statements**, 33
- **DBMOVER Statements**, 34
- **PowerExchange Message Log Options**, 163
- **Consumer API (CAPI) Connection Statements**, 166
- **DBMOVER Statements for Commands Issued through the pwxcmd Program**, 169
- **Sample DBMOVER Configuration Files**, 170

**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:

<table>
<thead>
<tr>
<th>Statement</th>
<th>Description</th>
<th>Reference</th>
</tr>
</thead>
<tbody>
<tr>
<td>ABEND_SW</td>
<td>Controls whether PowerExchange activates additional error routines to handle abends.</td>
<td>&quot;ABEND_SW Statement&quot; on page 34</td>
</tr>
<tr>
<td>ADA_L3_ALLOW</td>
<td>An Adabas optimization statement that controls whether PowerExchange uses Adabas L3 commands to read records from a file in logical sequence by descriptor value.</td>
<td>&quot;ADA_L3_ALLOW Statement&quot; on page 35</td>
</tr>
<tr>
<td>ADABAS_DEFAULT_DBID</td>
<td>Specifies the DBID value that PowerExchange uses when a data map specifies 0 in the <strong>Database ID</strong> property and the bulk data movement session does not specify an override value.</td>
<td>&quot;ADABAS_DEFAULT_DBID Statement&quot; on page 36</td>
</tr>
<tr>
<td>ADABAS_PREFIX</td>
<td>Specifies the prefix that PowerExchange uses to construct a user ID to access Adabas files.</td>
<td>&quot;ADABAS_PREFIX Statement&quot; on page 36</td>
</tr>
<tr>
<td>ADABASCODEPAGE</td>
<td>Specifies the single-byte and multibyte code pages to use for an Adabas database.</td>
<td>&quot;ADABASCODEPAGE Statement&quot; on page 37</td>
</tr>
<tr>
<td>ADAOPT</td>
<td>An Adabas optimization statement that controls whether PowerExchange uses Adabas L3 commands to read records from a file in logical sequence by descriptor value.</td>
<td>&quot;ADAOPT Statement&quot; on page 38</td>
</tr>
<tr>
<td>ADAOPTM</td>
<td>Deprecated. Exists for backward compatibility.</td>
<td>&quot;ADAOPTM Statement&quot; on page 39</td>
</tr>
<tr>
<td>ADAPREFETCH</td>
<td>Controls whether PowerExchange uses the Adabas prefetch feature when reading records.</td>
<td>&quot;ADAPREFETCH Statement&quot; on page 39</td>
</tr>
<tr>
<td>ADASTATS</td>
<td>Controls whether PowerExchange writes statistical information about Adabas operations to the PowerExchange message log file.</td>
<td>&quot;ADASTATS Statement&quot; on page 39</td>
</tr>
<tr>
<td>Statement</td>
<td>Description</td>
<td>Reference</td>
</tr>
<tr>
<td>-----------------</td>
<td>-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
<td>------------------------------------------------</td>
</tr>
<tr>
<td>ADAUSER</td>
<td>Controls whether PowerExchange uses the job name as the Adabas user ID to access Adabas files in a netport job.</td>
<td>“ADAUSER Statement” on page 40</td>
</tr>
<tr>
<td>APPBUFSIZE</td>
<td>Specifies the maximum data buffer size in bytes.</td>
<td>“APPBUFSIZE Statement” on page 40</td>
</tr>
<tr>
<td>APPBUFSIZEEDYN</td>
<td>For supported data sources, specifies whether to dynamically increase the maximum data buffer size.</td>
<td>“APPBUFSIZEEDYN Statement” on page 41</td>
</tr>
<tr>
<td>AS400EVENTMSGQ</td>
<td>Specifies an event message queue on an i5/OS system to which PowerExchange writes messages.</td>
<td>“AS400EVENTMSGQ Statement” on page 41</td>
</tr>
<tr>
<td>AS400USRJRNCODE</td>
<td>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.</td>
<td>“AS400USRJRNCODE Statement” on page 42</td>
</tr>
<tr>
<td>BS</td>
<td>Specifies the block size that PowerExchange uses when dynamically allocating target data sets on z/OS if you do not provide a block size.</td>
<td>“BS Statement” on page 43</td>
</tr>
<tr>
<td>CAPI_CONN_NAME</td>
<td>Specifies the default CAPI_CONNECTION statement for all data source types.</td>
<td>“CAPI_CONN_NAME Statement” on page 43</td>
</tr>
<tr>
<td>CAPI_CONNECTION - AS4J</td>
<td>Specifies the Consumer API (CAPI) parameters to use for DB2 for i5/OS CDC sources.</td>
<td>“CAPI_CONNECTION - AS4J Statement” on page 44</td>
</tr>
<tr>
<td>Statement</td>
<td>Description</td>
<td>Reference</td>
</tr>
<tr>
<td>------------------------</td>
<td>-----------------------------------------------------------------------------</td>
<td>-----------------------------------------------</td>
</tr>
<tr>
<td>CAPI_CONNECTION - CAPX</td>
<td>Specifies the CAPI parameters to use for continuous extraction of change data from PowerExchange Logger for Linux, UNIX, and Windows log files.</td>
<td>“CAPI_CONNECTION - CAPX Statement” on page 49</td>
</tr>
<tr>
<td>CAPI_CONNECTION - LRAP</td>
<td>Specifies the CAPI parameters to use for the Log Read API (LRAPI) component of the PowerExchange Logger for MVS.</td>
<td>“CAPI_CONNECTION - LRAP Statement” on page 51</td>
</tr>
<tr>
<td>CAPI_CONNECTION - MSQL</td>
<td>Specifies the CAPI parameters to use for Microsoft SQL Server CDC sources.</td>
<td>“CAPI_CONNECTION - MSQL Statement” on page 53</td>
</tr>
<tr>
<td>CAPI_CONNECTION - ORAD</td>
<td>Specifies the CAPI parameters to use for PowerExchange Express CDC for Oracle sources.</td>
<td>“CAPI_CONNECTION - ORAD Statement” on page 56</td>
</tr>
<tr>
<td>CAPI_CONNECTION - ORCL</td>
<td>Specifies the CAPI parameters to use for PowerExchange Oracle CDC with LogMiner sources.</td>
<td>“CAPI_CONNECTION - ORCL Statement” on page 58</td>
</tr>
<tr>
<td>CAPI_CONNECTION - UDB</td>
<td>Specifies the CAPI parameters to use for DB2 for Linux, UNIX, and Windows (LUW) CDC sources.</td>
<td>“CAPI_CONNECTION - UDB Statement” on page 64</td>
</tr>
<tr>
<td>CAPI_CONNECTION - UOWC</td>
<td>Specifies the CAPI parameters to use for the UOW Cleanser.</td>
<td>“CAPI_CONNECTION - UOWC Statement” on page 68</td>
</tr>
<tr>
<td>CAPI_SRC_DFLT</td>
<td>Specifies the default CAPI_CONNECTION statement for a specific data source type.</td>
<td>“CAPI_SRC_DFLT Statement” on page 73</td>
</tr>
<tr>
<td>CAPT_PATH</td>
<td>Specifies the path to the local directory on a Linux, UNIX, and Windows system that contains the control files for CDC.</td>
<td>“CAPT_PATH Statement” on page 74</td>
</tr>
<tr>
<td>CAPT_XTRA</td>
<td>Specifies the path to the local directory on a Linux, UNIX, and Windows system that stores extraction maps for CDC.</td>
<td>“CAPT_XTRA Statement” on page 75</td>
</tr>
<tr>
<td>Statement</td>
<td>Description</td>
<td>Reference</td>
</tr>
<tr>
<td>---------------</td>
<td>------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
<td>-------------------------------</td>
</tr>
<tr>
<td>CMDNODE</td>
<td>Specifies connection information for a PowerExchange Listener, PowerExchange Condense, or PowerExchange for Linux, UNIX, and Windows Logger process that is the target of pwxcmd commands.</td>
<td>“CMDNODE Statement” on page 75</td>
</tr>
<tr>
<td>CODEPAGE</td>
<td>Specifies the code pages that PowerExchange uses for operating system information, data, and literal values in SQL statements.</td>
<td>“CODEPAGE Statement” on page 76</td>
</tr>
<tr>
<td>COLON</td>
<td>Defines the character that PowerExchange uses as a the delimiter character for time values in times and time stamp fields.</td>
<td>“COLON Statement” on page 77</td>
</tr>
<tr>
<td>COMPRESS</td>
<td>Controls whether PowerExchange uses its proprietary compression routines to compress data that it transmits across the network.</td>
<td>“COMPRESS Statement” on page 78</td>
</tr>
<tr>
<td>CONSOLE_CODEPAGE</td>
<td>Specifies the code page that PowerExchange uses to display message text on consoles.</td>
<td>“CONSOLE_CODEPAGE Statement” on page 78</td>
</tr>
<tr>
<td>CONSOLE_TRACE</td>
<td>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.</td>
<td>“CONSOLE_TRACE Statement” on page 78</td>
</tr>
<tr>
<td>CONVCHAR</td>
<td>Controls how PowerExchange handles conversion errors for CHAR and VARCHAR columns that PowerExchange detects during bulk data movement read operations.</td>
<td>“CONVCHAR Statement” on page 79</td>
</tr>
<tr>
<td>Statement</td>
<td>Description</td>
<td>Reference</td>
</tr>
<tr>
<td>-------------------</td>
<td>-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
<td>-------------------------------------------------------</td>
</tr>
<tr>
<td>CPX_DIR</td>
<td>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.</td>
<td>“CPX_DIR Statement” on page 80</td>
</tr>
<tr>
<td>CREDENTIALS_CASE</td>
<td>Controls the case that PowerExchange uses for operating system user IDs and passwords.</td>
<td>“CREDENTIALS_CASE Statement” on page 80</td>
</tr>
<tr>
<td>DATAMAP_SERVER</td>
<td>Specifies the node name of the PowerExchange Listener that acts as the server for access requests to the file that stores data maps.</td>
<td>“DATAMAP_SERVER Statement” on page 81</td>
</tr>
<tr>
<td>DATERANGE</td>
<td>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.</td>
<td>“DATERANGE Statement” on page 82</td>
</tr>
<tr>
<td>DB2_BIN_AS_CHAR</td>
<td>Controls whether PowerExchange considers data in CHAR and VARCHAR columns that you define with the FOR BIT DATA clause as character data.</td>
<td>“DB2_BIN_AS_CHAR Statement” on page 82</td>
</tr>
<tr>
<td>DB2_BIN_CODEPAGE</td>
<td>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.</td>
<td>“DB2_BIN_CODEPAGE Statement” on page 83</td>
</tr>
<tr>
<td>DB2_ERRORFILE</td>
<td>Specifies the name of the user-customized SQL error file that PowerExchange uses for DB2 bulk data movement operations.</td>
<td>“DB2_ERRORFILE Statement” on page 83</td>
</tr>
<tr>
<td>DB2CODEPAGE</td>
<td>Defines the single-byte, mixed, and graphics CCSID values that PowerExchange uses to process bulk data from the specified DB2 subsystem.</td>
<td>“DB2CODEPAGE Statement” on page 84</td>
</tr>
<tr>
<td>Statement</td>
<td>Description</td>
<td>Reference</td>
</tr>
<tr>
<td>-----------------------</td>
<td>-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
<td>-----------------------------------------------</td>
</tr>
<tr>
<td>DB2DEF_ENCODING</td>
<td>Defines the default encoding scheme that PowerExchange assigns to any DB2 columns without an encoding scheme when you create a DB2 data map.</td>
<td>“DB2DEF_ENCODING Statement” on page 88</td>
</tr>
<tr>
<td>DB2ID</td>
<td>Defines the DB2 plan and the PowerExchange access method module that PowerExchange uses to process bulk data from the specified DB2 subsystem.</td>
<td>“DB2ID Statement” on page 89</td>
</tr>
<tr>
<td>DB2PLAN</td>
<td>Defines the DB2 plan that PowerExchange uses for bulk data movement processing.</td>
<td>“DB2PLAN Statement” on page 89</td>
</tr>
<tr>
<td>DECPONIT</td>
<td>Defines the character that PowerExchange uses as the decimal point character in fields that contain noninteger numbers.</td>
<td>“DECPONIT Statement” on page 90</td>
</tr>
<tr>
<td>DEFAULTCHAR</td>
<td>Defines the character that PowerExchange uses to replace any character data in a nonrelational data source that is not valid.</td>
<td>“DEFAULTCHAR Statement” on page 90</td>
</tr>
<tr>
<td>DEFAULTDATE</td>
<td>Defines a numeric date string that PowerExchange uses to replace any missing component in a date field.</td>
<td>“DEFAULTDATE Statement” on page 91</td>
</tr>
<tr>
<td>DISABLE_PARTITIONS</td>
<td>Controls whether PowerExchange disables partitioning of the source data for bulk data movement operations.</td>
<td>“DISABLE_PARTITIONS Statement” on page 91</td>
</tr>
<tr>
<td>DISP</td>
<td>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.</td>
<td>“DISP Statement” on page 91</td>
</tr>
<tr>
<td>Statement</td>
<td>Description</td>
<td>Reference</td>
</tr>
<tr>
<td>---------------------------</td>
<td>-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
<td>-----------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>DMRESOURCE</td>
<td>Specifies the suffix that PowerExchange uses to create the PowerExchange-specific resource profile that controls access to the DATAMAPS data set.</td>
<td>“DMRESOURCE Statement” on page 92</td>
</tr>
<tr>
<td>DM_SUBTASK</td>
<td>Controls how the PowerExchange Listener accesses the file that stores data maps.</td>
<td>“DM_SUBTASK Statement” on page 93</td>
</tr>
<tr>
<td>DMXCACHE_DELETEECSA</td>
<td>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</td>
<td>“DMXCACHE_DELETEECSA Statement” on page 94</td>
</tr>
<tr>
<td>DMXCACHE_MAX_MEMORY_MB</td>
<td>Enables PowerExchange data maps caching and configures the maximum size for the data maps cache in megabytes.</td>
<td>“DMXCACHE_MAX_MEMORY_MB Statement” on page 94</td>
</tr>
<tr>
<td>DMXCACHE_MULTIPLEJOBS</td>
<td>Determines whether PowerExchange runs data maps caching in multiple-jobs or single-job mode.</td>
<td>“DMXCACHE_MULTIPLEJOBS Statement” on page 95</td>
</tr>
<tr>
<td>DMX_DIR</td>
<td>Specifies the location that PowerExchange uses to read and store data maps.</td>
<td>“DMX_DIR Statement” on page 96</td>
</tr>
<tr>
<td>DTLMSG_CODEPAGE</td>
<td>Specifies the code page identifier that indicates which version of the PowerExchange messages file to use.</td>
<td>“DTLMSG_CODEPAGE Statement” on page 96</td>
</tr>
<tr>
<td>ENCRYPT</td>
<td>Controls whether PowerExchange uses encryption when moving data. In addition to an internal proprietary encryption method, PowerExchange supports DES and RSA Security RC2 encryption.</td>
<td>“ENCRYPT Statement” on page 97</td>
</tr>
<tr>
<td>Statement</td>
<td>Description</td>
<td>Reference</td>
</tr>
<tr>
<td>-----------------</td>
<td>-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
<td>------------------------------------------------</td>
</tr>
<tr>
<td>ENCRYPTLEVEL</td>
<td>Specifies the encryption level to use if you select DES or RSA Security RC2 encryption. The encryption level determines the encryption key length. If you enter DES or RC2 in the ENCRYPT statement, you must include the ENCRYPTLEVEL statement.</td>
<td>“ENCRYPTLEVEL Statement” on page 98</td>
</tr>
<tr>
<td>ENQMAJORNAME</td>
<td>Specifies the major name for ENQ macros that PowerExchange issues to lock PowerExchange data sets and resources.</td>
<td>“ENQMAJORNAME Statement” on page 99</td>
</tr>
<tr>
<td>ENQSYSTEMS</td>
<td>Controls whether PowerExchange uses a scope of SYSTEM or SYSTEMS for ENQ macros that it issues to lock PowerExchange data sets and resources.</td>
<td>“ENQSYSTEMS Statement” on page 99</td>
</tr>
<tr>
<td>ERRROWNOTFOUND</td>
<td>Specifies whether PowerExchange generates or does not generate errors for UPDATE or DELETE operations on nonexistent rows.</td>
<td>“ERRROWNOTFOUND Statement” on page 100</td>
</tr>
<tr>
<td>EXT_CP_SUPPT</td>
<td>Controls character conversion in single-byte static code pages.</td>
<td>“EXT_CP_SUPPT Statement” on page 100</td>
</tr>
<tr>
<td>GDGLOCATE</td>
<td>Controls whether PowerExchange instructs dynamic allocation to use a LOCATE to get the most recent catalog information for generation data set allocation requests.</td>
<td>“GDGLOCATE Statement” on page 101</td>
</tr>
<tr>
<td>GSBUFSIZE</td>
<td>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.</td>
<td>“GSBUFIZE Statement” on page 102</td>
</tr>
<tr>
<td>ICUALIAS</td>
<td>Adds or deletes an alias that points to a custom ICU code page.</td>
<td>“ICUALIAS Statement” on page 102</td>
</tr>
<tr>
<td>Statement</td>
<td>Description</td>
<td>Reference</td>
</tr>
<tr>
<td>----------------------</td>
<td>-----------------------------------------------------------------------------</td>
<td>---------------------------------------------------------</td>
</tr>
<tr>
<td>ICUCNVPROPERTY</td>
<td>Sets customized properties for an ICU code page.</td>
<td>“ICUCNVPROPERTY Statement” on page 103</td>
</tr>
<tr>
<td>ICUCONVERTER</td>
<td>Adds a code page converter to or deletes a code page converter from an ICU code page.</td>
<td>“ICUCONVERTER Statement” on page 105</td>
</tr>
<tr>
<td>ICUDATADIR</td>
<td>Defines the directory location for the binary CNV files that makeconv produces.</td>
<td>“ICUDATADIR Statement” on page 107</td>
</tr>
<tr>
<td>IMSID</td>
<td>Defines the IMS subsystem information that PowerExchange uses for IMS bulk data movement operations and for IMS log-based CDC.</td>
<td>“IMSID Statement” on page 107</td>
</tr>
<tr>
<td>JOBCLASS</td>
<td>Specifies the job class that PowerExchange uses for CA IDMS/DB metadata and DB2 LOAD utility batch jobs.</td>
<td>“JOBCLASS Statement” on page 108</td>
</tr>
<tr>
<td>LISTENER</td>
<td>Defines the TCP/IP port on which a named PowerExchange Listener process listens for work requests.</td>
<td>“LISTENER Statement” on page 109</td>
</tr>
<tr>
<td>LOADCTLFILE</td>
<td>Specifies the PDS data set that contains the control card template member for DB2 for z/OS LOAD utility batch jobs.</td>
<td>“LOADCTLFILE Statement” on page 110</td>
</tr>
<tr>
<td>LOADJOBFILE</td>
<td>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.</td>
<td>“LOADJOBFILE Statement” on page 111</td>
</tr>
<tr>
<td>LOG_CODEPAGE</td>
<td>Specifies the code page that PowerExchange uses to write messages to the log file.</td>
<td>“LOG_CODEPAGE Statement” on page 111</td>
</tr>
<tr>
<td>LOG_LINE_LIMIT</td>
<td>Specifies the maximum line length for PowerExchange messages in the message log.</td>
<td>“LOG_LINE_LIMIT Statement” on page 112</td>
</tr>
<tr>
<td>Statement</td>
<td>Description</td>
<td>Reference</td>
</tr>
<tr>
<td>---------------</td>
<td>-----------------------------------------------------------------------------</td>
<td>-----------------------------------------------</td>
</tr>
<tr>
<td>LOGPATH</td>
<td>Defines the directory location where PowerExchange writes message log files on Linux, UNIX, and Windows systems.</td>
<td>“LOGPATH Statement” on page 112</td>
</tr>
<tr>
<td>LOGSID</td>
<td>Specifies the location of the CA IDMS/DB logs and the PowerExchange log catalog.</td>
<td>“LOGSID Statement” on page 113</td>
</tr>
<tr>
<td>LOWVALUES</td>
<td>Specifies whether PowerExchange preserves hexadecimal '0' values in a VSAM or sequential data source when writing to a VSAM or sequential target.</td>
<td>“LOWVALUES Statement” on page 113</td>
</tr>
<tr>
<td>LRECL</td>
<td>Specifies the logical record length that PowerExchange uses when dynamically allocating target data sets on z/OS, if you do not provide LRECL information.</td>
<td>“LRECL Statement” on page 114</td>
</tr>
<tr>
<td>MAXTASKS</td>
<td>Defines the maximum number of tasks that can run concurrently in a PowerExchange Listener.</td>
<td>“MAXTASKS Statement” on page 114</td>
</tr>
<tr>
<td>MSGPREFIX</td>
<td>Defines the string that PowerExchange uses to prefix PowerExchange messages.</td>
<td>“MSGPREFIX Statement” on page 115</td>
</tr>
<tr>
<td>MSGPREFIX-HYPHEN</td>
<td>Specifies whether PowerExchange includes or excludes the hyphen character between the message prefix and the message number.</td>
<td>“MSGPREFIX-HYPHEN Statement” on page 115</td>
</tr>
<tr>
<td>MSS_ERRORFILE</td>
<td>Specifies the name of the user-customized SQL error file that PowerExchange uses for Microsoft SQL Server bulk data movement operations.</td>
<td>“MSS_ERRORFILE Statement” on page 116</td>
</tr>
<tr>
<td>MVSD2BAF</td>
<td>Specifies which DB2 for z/OS attachment facility PowerExchange uses for DB2 bulk data movement operations.</td>
<td>“MVSD2BAF Statement” on page 116</td>
</tr>
<tr>
<td>Statement</td>
<td>Description</td>
<td>Reference</td>
</tr>
<tr>
<td>-------------------</td>
<td>-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
<td>-----------------------------------------------------------------</td>
</tr>
<tr>
<td>NEGSIGN</td>
<td>Defines the character that PowerExchange uses as the negative sign character in fields that contain negative numbers.</td>
<td>&quot;NEGSIGN Statement&quot; on page 117</td>
</tr>
<tr>
<td>NETPORT</td>
<td>Defines information about netport jobs and associates the netport job with a specific listener port.</td>
<td>&quot;NETPORT Statement&quot; on page 117</td>
</tr>
<tr>
<td>NODE</td>
<td>Defines the TCP/IP host name and port that PowerExchange uses to contact a PowerExchange Listener process.</td>
<td>&quot;NODE Statement&quot; on page 120</td>
</tr>
<tr>
<td>NOGETHOSTBYNAME</td>
<td>Controls whether PowerExchange gets the IP address of the local system to verify the license and include it in PowerExchange messages.</td>
<td>&quot;NOGETHOSTBYNAME Statement&quot; on page 121</td>
</tr>
<tr>
<td>NRDB_WRITE_CHAR_NULL_FILL</td>
<td>Defines the character or hexadecimal value that PowerExchange uses to replace null characters in a field.</td>
<td>&quot;NRDB_WRITE_CHAR_NULL_FILL Statement&quot; on page 122</td>
</tr>
<tr>
<td>NRDB_WRITE_NUM_NULL_FILL</td>
<td>Defines the numeric or hexadecimal value that PowerExchange uses to replace null values in an unpacked numeric field.</td>
<td>&quot;NRDB_WRITE_NUM_NULL_FILL Statement&quot; on page 122</td>
</tr>
<tr>
<td>NUMERICSIGN</td>
<td>Controls whether PowerExchange delivers unsigned fields as a positive 'X'C' or as unsigned 'XF'.</td>
<td>&quot;NUMERICSIGN Statement&quot; on page 123</td>
</tr>
<tr>
<td>ODBASUPP</td>
<td>Controls whether PowerExchange can use the Open Database Access (ODBA) interface to access IMS databases and unload data sets for bulk data movement operations.</td>
<td>&quot;ODBASUPP Statement&quot; on page 123</td>
</tr>
<tr>
<td>ORA_ERRORFILE</td>
<td>Specifies the name of the user-customized SQL error file that PowerExchange uses for Oracle bulk data movement operations.</td>
<td>&quot;ORA_ERRORFILE Statement&quot; on page 123</td>
</tr>
<tr>
<td>Statement</td>
<td>Description</td>
<td>Reference</td>
</tr>
<tr>
<td>------------------------</td>
<td>-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
<td>-----------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>ORACLE_CAPTURE_TYPE</td>
<td>Specifies which PowerExchange Oracle CDC solution is in use for a PowerExchange installation: PowerExchange Express CDC for Oracle or PowerExchange Oracle LogMiner CDC.</td>
<td>“ORACLE_CAPTURE_TYPE Statement” on page 124</td>
</tr>
</tbody>
</table>
| ORACLE_UNHANDLED_NUMASCHAR | Controls whether PowerExchange handles the following types of numeric columns in Oracle source tables in a manner that prevents data loss:  
- 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 125                                          |
<p>| 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 126                                                       |
| ORACLEID               | Specifies the Oracle source database and connection information for PowerExchange Oracle CDC with LogMiner and PowerExchange Express CDC for Oracle.                                | “ORACLEID Statement” on page 127                                                             |
| OUSP                   | Loads the z/OS OpenEdition security environment when using ADABAS. Use at the direction of Informatica Global Customer Support.                                                                          | “OUSP Statement” on page 129                                                                 |
| 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 130                                                                 |</p>
<table>
<thead>
<tr>
<th>Statement</th>
<th>Description</th>
<th>Reference</th>
</tr>
</thead>
<tbody>
<tr>
<td>PIPE</td>
<td>Defines the character that PowerExchange uses to separate fields in a TXT output file.</td>
<td>“PIPE Statement” on page 130</td>
</tr>
<tr>
<td>POLLTIME</td>
<td>Defines the amount of time, in thousandths of a second, that the PowerExchange Listener waits before polling for outstanding connections.</td>
<td>“POLLTIME Statement” on page 130</td>
</tr>
<tr>
<td>PRGIND</td>
<td>Controls whether PowerExchange writes or does not write read progress messages to the PowerExchange message log file.</td>
<td>“PRGIND Statement” on page 131</td>
</tr>
<tr>
<td>PRGINT</td>
<td>Defines the interval after which PowerExchange writes read progress messages to the PowerExchange message log file, if requested.</td>
<td>“PRGINT Statement” on page 131</td>
</tr>
<tr>
<td>PWXSOMAXCONN</td>
<td>Specifies the maximum number of TCP/IP socket connections that a PowerExchange Listener uses to listen for work.</td>
<td>“PWXSOMAXCONN Statement” on page 131</td>
</tr>
<tr>
<td>RACF_CLASS</td>
<td>Specifies the RACF class name that PowerExchange uses when checking PowerExchange-specific resource profiles.</td>
<td>“RACF_CLASS Statement” on page 132</td>
</tr>
<tr>
<td>RDBMSINSRTDFLT</td>
<td>Controls whether PowerExchange uses DB2 default values for DB2 columns that you define with the WITH DEFAULT clause.</td>
<td>“RDBMSINSRTDFLT Statement” on page 132</td>
</tr>
<tr>
<td>RECFM</td>
<td>Specifies the record format that PowerExchange uses when dynamically allocating target data sets on z/OS, if you do not provide RECFM information.</td>
<td>“RECFM Statement” on page 133</td>
</tr>
<tr>
<td>REJECT_FILE_DELIMITER</td>
<td>Defines the character that PowerExchange uses as the default delimiter character in a reject file.</td>
<td>“REJECT_FILE_DELIMITER Statement” on page 133</td>
</tr>
<tr>
<td>Statement</td>
<td>Description</td>
<td>Reference</td>
</tr>
<tr>
<td>------------</td>
<td>--------------------------------------------------------------------------------------------------</td>
<td>-------------------------------------</td>
</tr>
<tr>
<td>RELEASE</td>
<td>Controls whether PowerExchange requests or does not request to release unused space for data sets that it dynamically allocates.</td>
<td>“RELEASE Statement” on page 134</td>
</tr>
<tr>
<td>RMTRDBDIRE</td>
<td>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.</td>
<td>“RMTRDBDIRE Statement” on page 134</td>
</tr>
<tr>
<td>RMTSYSNAME</td>
<td>Defines the name of the i5/OS host system that contains the DB2 source tables for CDC, local journals, and journal receivers.</td>
<td>“RMTSYSNAME Statement” on page 134</td>
</tr>
<tr>
<td>SECURITY</td>
<td>Controls whether PowerExchange completes user authentication, and controls access to resources and commands.</td>
<td>“SECURITY Statement” on page 195</td>
</tr>
<tr>
<td>SERVICE_TIMEOUT</td>
<td>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.</td>
<td>“SERVICE_TIMEOUT Statement” on page 138</td>
</tr>
<tr>
<td>SESSID</td>
<td>Specifies the value to use as the DB2 Correlation ID for DB2 requests.</td>
<td>“SESSID Statement” on page 138</td>
</tr>
<tr>
<td>SHOW_THREAD_PERF</td>
<td>Specifies the number of records PowerExchange processes before writing statistics messages about multithreaded processing to the PowerExchange message log file.</td>
<td>“SHOW_THREAD_PERF Statement” on page 139</td>
</tr>
<tr>
<td>SPACE</td>
<td>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.</td>
<td>“SPACE Statement” on page 139</td>
</tr>
<tr>
<td>Statement</td>
<td>Description</td>
<td>Reference</td>
</tr>
<tr>
<td>----------------------------</td>
<td>-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
<td>------------------------------------------------</td>
</tr>
<tr>
<td>SSL</td>
<td>Specifies SSL certificate information for a Secure Sockets Layer (SSL) connection.</td>
<td>“SSL Statement” on page 140</td>
</tr>
<tr>
<td>SSL_ALLOW_SELFSIGNED</td>
<td>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.</td>
<td>“SSL_ALLOW_SELFSIGNED Statement” on page 141</td>
</tr>
<tr>
<td>SSL_CIPHER_LIST</td>
<td>Restricts the available ciphers that a client offers to a server during an SSL handshake to the specified list.</td>
<td>“SSL_CIPHER_LIST Statement” on page 141</td>
</tr>
<tr>
<td>SSL_CONTEXT_METHOD</td>
<td>Selects the SSL or TLS versions that the peer supports for PowerExchange SSL communication.</td>
<td>“SSL_CONTEXT_METHOD Statement” on page 142</td>
</tr>
<tr>
<td>SSL_REQ_CLNT_CERT</td>
<td>Controls whether a PowerExchange server authenticates the identity of a PowerExchange client by using an SSL certificate.</td>
<td>“SSL_REQ_CLNT_CERT Statement” on page 143</td>
</tr>
<tr>
<td>SSL_REQ_SRVR_CERT</td>
<td>Controls whether a PowerExchange client authenticates the identity of a PowerExchange server by using an SSL certificate.</td>
<td>“SSL_REQ_SRVR_CERT Statement” on page 143</td>
</tr>
<tr>
<td>STATS</td>
<td>Controls whether PowerExchange writes SMF statistics records for the PowerExchange Listener to SMF or to a file.</td>
<td>“STATS Statement” on page 144</td>
</tr>
<tr>
<td>SUBMITTIMEOUT</td>
<td>Specifies the time, in seconds, that a PowerExchange Listener waits to receive notification from a spawned batch job that it has started.</td>
<td>“SUBMITTIMEOUT Statement” on page 145</td>
</tr>
<tr>
<td>SUP_FUNC</td>
<td>Defines the function name of the zIIP enclave for the PowerExchange Listener to the IBM Workload Manager.</td>
<td>“SUP_FUNC Statement” on page 146</td>
</tr>
<tr>
<td>Statement</td>
<td>Description</td>
<td>Reference</td>
</tr>
<tr>
<td>------------------------</td>
<td>------------------------------------------------------------------------------</td>
<td>------------------------------------------------</td>
</tr>
<tr>
<td>SUP_REQUEST</td>
<td>Defines the name of the requestor for zIIP facilities.</td>
<td>“SUP_REQUEST Statement” on page 146</td>
</tr>
<tr>
<td>SUP_SSNAME</td>
<td>Defines the subsystem name that identifies the PowerExchange Listener started task to the IBM Workload Manager when zIIP exploitation is enabled.</td>
<td>“SUP_SSNAME Statement” on page 146</td>
</tr>
<tr>
<td>SUP_SSTYPE</td>
<td>Defines the name that the IBM Workload Manager uses as the subsystem type for the enclave for zIIP.</td>
<td>“SUP_SSTYPE Statement” on page 147</td>
</tr>
<tr>
<td>SUPPRESS_DATA_LOGGING</td>
<td>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.</td>
<td>“SUPPRESS_DATA_LOGGING Statement” on page 147</td>
</tr>
<tr>
<td>SVCNODE</td>
<td>Specifies the TCP/IP port on which a PowerExchange Listener or PowerExchange Logger for Linux, UNIX, and Windows process listens for commands.</td>
<td>“SVCNODE Statement” on page 147</td>
</tr>
<tr>
<td>SYSOUT_TIMESTAMP</td>
<td>Controls whether PowerExchange includes time stamp information in output files and console output.</td>
<td>“SYSOUT_TIMESTAMP Statement” on page 149</td>
</tr>
<tr>
<td>TAPEWAIT</td>
<td>Controls whether netport jobs wait or do not wait for tape volumes.</td>
<td>“TAPEWAIT Statement” on page 149</td>
</tr>
<tr>
<td>TCPIP_DIAGNOSTICS_TRACE</td>
<td>Controls the issuing of network diagnostic messages PWX-33316, PWX-33317, and PWX-33318.</td>
<td>“TCPIP_DIAGNOSTICS_TRACE Statement” on page 149</td>
</tr>
<tr>
<td>TCPIP_DIAGNOSTICS_TRACE_SZ</td>
<td>Controls the number of trace records to keep for network send and receive operations.</td>
<td>“TCPIP_DIAGNOSTICS_TRACE_SZ Statement” on page 150</td>
</tr>
<tr>
<td>TCPIPBUFSIZE</td>
<td>Deprecated. Exists for backward compatibility.</td>
<td>“TCPIPBUFSIZE Statement” on page 151</td>
</tr>
<tr>
<td>Statement</td>
<td>Description</td>
<td>Reference</td>
</tr>
<tr>
<td>------------------</td>
<td>-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
<td>----------------------------------------</td>
</tr>
<tr>
<td>TCPIPVER</td>
<td>Specifies alternative TCP/IP socket code that PowerExchange uses instead of the standard IBM TCP/IP UNIX socket code.</td>
<td>“TCPIPVER Statement” on page 151</td>
</tr>
<tr>
<td>TEMPHLQ</td>
<td>Overrides the high-level qualifier that PowerExchange uses by default when creating a temporary file for CA IDMS/DB metadata.</td>
<td>“TEMPHLQ Statement” on page 151</td>
</tr>
<tr>
<td>TEXT_EOF_FOR_BINARY</td>
<td>Controls whether PowerExchange ignores hexadecimal value 1A when reading binary flat files on Linux, UNIX, or Windows.</td>
<td>“TEXT_EOF_FOR_BINARY Statement” on page 152</td>
</tr>
<tr>
<td>TIMEZONE</td>
<td>Specifies a local time zone that PowerExchange uses for reading or writing Adabas date-time values for bulk data movement sessions.</td>
<td>“TIMEZONE Statement” on page 152</td>
</tr>
<tr>
<td>TRACE</td>
<td>Activates traces in PowerExchange code paths for diagnostic purposes.</td>
<td>“TRACE Statement” on page 153</td>
</tr>
<tr>
<td>TRACING</td>
<td>Activates PowerExchange alternative logging and specifies attributes for the alternative log files.</td>
<td>“TRACING Statement” on page 153</td>
</tr>
<tr>
<td>UNIT</td>
<td>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.</td>
<td>“UNIT Statement” on page 159</td>
</tr>
<tr>
<td>USE_TYPE1_FALLBACKS</td>
<td>Enables type 1 fallback mappings for all code pages.</td>
<td>“USE_TYPE1_FALLBACKS Statement” on page 159</td>
</tr>
<tr>
<td>USESUP</td>
<td>Controls whether PowerExchange uses zIIP for zIIP-enabled PowerExchange Listener functions.</td>
<td>“USESUP Statement” on page 160</td>
</tr>
<tr>
<td>Statement</td>
<td>Description</td>
<td>Reference</td>
</tr>
<tr>
<td>--------------------</td>
<td>------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
<td>--------------------</td>
</tr>
<tr>
<td>VOLSER</td>
<td>Specifies the volume serial number that PowerExchange uses when dynamically allocating target data sets on z/OS, if you do not provide VOLSER information.</td>
<td>“VOLSER Statement” on page 160</td>
</tr>
<tr>
<td>VSAM</td>
<td>Specifies the number of buffers that PowerExchange uses for data and index control intervals when processing VSAM data sets.</td>
<td>“VSAM Statement” on page 161</td>
</tr>
<tr>
<td>WAITDSN</td>
<td>Controls whether netport jobs wait for in-use data sets.</td>
<td>“WAITDSN Statement” on page 162</td>
</tr>
<tr>
<td>WORKCLASS</td>
<td>Defines the transaction name for Workload Manager classification when z/IPP exploitation is enabled.</td>
<td>“WORKCLASS Statement” on page 162</td>
</tr>
<tr>
<td>WRT_ERROR_HANDLING</td>
<td>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.</td>
<td>“WRT_ERROR_HANDLING Statement” on page 162</td>
</tr>
</tbody>
</table>

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:

<table>
<thead>
<tr>
<th>Statement</th>
<th>Description</th>
<th>Reference</th>
</tr>
</thead>
<tbody>
<tr>
<td>ADA_L3_ALLOW</td>
<td>An Adabas optimization statement that controls whether PowerExchange uses Adabas L3 commands to read records from a file in logical sequence by descriptor value.</td>
<td>&quot;ADA_L3_ALLOW Statement&quot; on page 36</td>
</tr>
<tr>
<td>ADABAS_DEFAULT_DBID</td>
<td>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.</td>
<td>&quot;ADABAS_DEFAULT_DBID Statement&quot; on page 36</td>
</tr>
<tr>
<td>ADABAS_PREFI</td>
<td>Specifies the prefix that PowerExchange uses to construct a user ID to access Adabas files.</td>
<td>&quot;ADABAS_PREFIX Statement&quot; on page 36</td>
</tr>
<tr>
<td>ADABASCODEPAGE</td>
<td>Specifies the single-byte and multibyte code pages to use for an Adabas database.</td>
<td>&quot;ADABASCODEPAGE Statement&quot; on page 37</td>
</tr>
<tr>
<td>ADAOPT</td>
<td>An Adabas optimization statement that controls whether PowerExchange uses Adabas L3 commands to read records from a file in logical sequence by descriptor value.</td>
<td>&quot;ADAOPT Statement&quot; on page 38</td>
</tr>
<tr>
<td>ADAOPTM</td>
<td>Deprecated. Exists for backward compatibility.</td>
<td>&quot;ADAOPTM Statement&quot; on page 39</td>
</tr>
<tr>
<td>ADAPREFETCH</td>
<td>Controls whether PowerExchange uses the Adabas prefetch feature when reading records.</td>
<td>&quot;ADAPREFETCH Statement&quot; on page 39</td>
</tr>
<tr>
<td>ADASTATS</td>
<td>Controls whether PowerExchange writes statistical information about Adabas operations to the PowerExchange message log file.</td>
<td>&quot;ADASTATS Statement&quot; on page 39</td>
</tr>
<tr>
<td>ADAUSER</td>
<td>Controls whether PowerExchange uses the job name as the Adabas user ID to access Adabas files in a netport job.</td>
<td>&quot;ADAUSER Statement&quot; on page 40</td>
</tr>
<tr>
<td>OUSP</td>
<td>Loads the z/OS OpenEdition security environment when using ADABAS. Use at the direction of Informatica Global Customer Support.</td>
<td>&quot;OUSP Statement&quot; on page 126</td>
</tr>
<tr>
<td>TIMEZONE</td>
<td>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.</td>
<td>&quot;TIMEZONE Statement&quot; on page 152</td>
</tr>
</tbody>
</table>

**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:

<table>
<thead>
<tr>
<th>Statement</th>
<th>Description</th>
<th>Reference</th>
</tr>
</thead>
<tbody>
<tr>
<td>LOADJOBFILE</td>
<td>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.</td>
<td>&quot;LOADJOBFILE Statement&quot; on page 111</td>
</tr>
<tr>
<td>LOGSID</td>
<td>Specifies the location of the CA IDMS/DB logs and the PowerExchange log catalog.</td>
<td>&quot;LOGSID Statement&quot; on page 113</td>
</tr>
<tr>
<td>PC_AUTH</td>
<td>Controls whether the PowerExchange Listener uses its MVS Program Call (PC) services routine to acquire the authorization to access CA IDMS/DB.</td>
<td>&quot;PC_AUTH Statement&quot; on page 130</td>
</tr>
<tr>
<td>TEMPHLQ</td>
<td>Overrides the high-level qualifier that PowerExchange uses by default when creating a temporary file for CA IDMS/DB metadata.</td>
<td>&quot;TEMPHLQ Statement&quot; on page 151</td>
</tr>
</tbody>
</table>

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:

<table>
<thead>
<tr>
<th>Statement</th>
<th>Description</th>
<th>Reference</th>
</tr>
</thead>
<tbody>
<tr>
<td>AS400USRJRNCODE</td>
<td>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.</td>
<td>&quot;AS400USRJRNCODE Statement&quot; on page 42</td>
</tr>
<tr>
<td>CAPI_CONNECTION - AS4J</td>
<td>Specifies the Consumer API (CAPI) parameters needed for DB2 for i5/OS CDC sources.</td>
<td>&quot;CAPI_CONNECTION - AS4J Statement&quot; on page 44</td>
</tr>
<tr>
<td>DB2_BIN_AS_CHAR</td>
<td>Controls whether PowerExchange considers data in CHAR and VARCHAR columns that you define with the FOR BIT DATA clause as character data.</td>
<td>&quot;DB2_BIN_AS_CHAR Statement&quot; on page 82</td>
</tr>
<tr>
<td>DB2_BIN_CODEPAGE</td>
<td>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.</td>
<td>&quot;DB2_BIN_CODEPAGE Statement&quot; on page 83</td>
</tr>
<tr>
<td>DB2_ERRORFILE</td>
<td>Specifies the name of the user-customized SQL error file that PowerExchange uses for DB2 bulk data movement operations.</td>
<td>&quot;DB2_ERRORFILE Statement&quot; on page 83</td>
</tr>
<tr>
<td>Statement</td>
<td>Description</td>
<td>Reference</td>
</tr>
<tr>
<td>-----------------</td>
<td>------------------------------------------------------------------------------------------------</td>
<td>-----------------------------------------------</td>
</tr>
<tr>
<td>RMTRDBDIRE</td>
<td>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.</td>
<td>“RMTRDBDIRE Statement” on page 134</td>
</tr>
<tr>
<td>RMTSYSNAME</td>
<td>Defines the name of the i5/OS host system that contains the DB2 source tables for CDC, local journals, and journal receivers.</td>
<td>“RMTSYSNAME Statement” on page 134</td>
</tr>
</tbody>
</table>

### 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:

<table>
<thead>
<tr>
<th>Statement</th>
<th>Description</th>
<th>Reference</th>
</tr>
</thead>
<tbody>
<tr>
<td>CAPI_CONNECTION</td>
<td>Specifies the CAPI parameters for DB2 for Linux, UNIX, and Windows CDC sources.</td>
<td>“CAPI_CONNECTION Statement” on page 64</td>
</tr>
<tr>
<td>DB2_ERRORFILE</td>
<td>Specifies the name of the user-customized SQL error file that PowerExchange uses for DB2 bulk data movement operations.</td>
<td>“DB2_ERRORFILE Statement” on page 83</td>
</tr>
</tbody>
</table>

### 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:

<table>
<thead>
<tr>
<th>Statement</th>
<th>Description</th>
<th>Reference</th>
</tr>
</thead>
<tbody>
<tr>
<td>DB2_ERRORFILE</td>
<td>Specifies the name of the user-customized SQL error file that PowerExchange uses for DB2 bulk data movement operations.</td>
<td>“DB2_ERRORFILE Statement” on page 83</td>
</tr>
<tr>
<td>DB2CODEPAGE</td>
<td>Defines the single-byte, mixed, and graphics CCSID values that PowerExchange uses to process bulk data from the specified DB2 subsystem.</td>
<td>“DB2CODEPAGE Statement” on page 84</td>
</tr>
<tr>
<td>DB2DEF_ENCODING</td>
<td>Defines the default encoding scheme that PowerExchange assigns to any DB2 columns without an encoding scheme when you create a DB2 data map.</td>
<td>“DB2DEF_ENCODING Statement” on page 86</td>
</tr>
<tr>
<td>DB2ID</td>
<td>Defines the DB2 plan and the PowerExchange access method module that PowerExchange uses to process bulk data from the specified DB2 subsystem.</td>
<td>“DB2ID Statement” on page 89</td>
</tr>
</tbody>
</table>
### 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:

<table>
<thead>
<tr>
<th>Statement</th>
<th>Description</th>
<th>Reference</th>
</tr>
</thead>
<tbody>
<tr>
<td>IMSID</td>
<td>Defines the IMS subsystem information that PowerExchange uses for IMS bulk data movement operations and for IMS log-based CDC.</td>
<td>“IMSID Statement” on page 107</td>
</tr>
<tr>
<td>ODBASUPP</td>
<td>Controls whether PowerExchange can use the Open Database Access (ODBA) interface to access IMS databases and unload data sets for bulk data movement operations.</td>
<td>“ODBASUPP Statement” on page 123</td>
</tr>
</tbody>
</table>

### 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:

<table>
<thead>
<tr>
<th>Statement</th>
<th>Description</th>
<th>Reference</th>
</tr>
</thead>
<tbody>
<tr>
<td>CAPI_CONNECTION - MSQML</td>
<td>Specifies the CAPI parameters for Microsoft SQL Server CDC sources.</td>
<td>“CAPI_CONNECTION - MSQML Statement” on page 53</td>
</tr>
<tr>
<td>MSS_ERRORFILE</td>
<td>Specifies the name of the user-customized SQL error file that PowerExchange uses for Microsoft SQL Server bulk data movement operations.</td>
<td>“MSS_ERRORFILE Statement” on page 116</td>
</tr>
</tbody>
</table>
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:

<table>
<thead>
<tr>
<th>Statement</th>
<th>Description</th>
<th>Reference</th>
</tr>
</thead>
<tbody>
<tr>
<td>DEFAULTCHAR</td>
<td>Defines the character that PowerExchange uses to replace any character data in a nonrelational data source that is not valid.</td>
<td>“DEFAULTCHAR Statement” on page 90</td>
</tr>
<tr>
<td>NRDB_WRITE_CHAR_NULL_FILL</td>
<td>Defines the character or hexadecimal value that PowerExchange uses to replace null characters in a field.</td>
<td>“NRDB_WRITE_CHAR_NULL_FILL Statement” on page 122</td>
</tr>
<tr>
<td>NRDB_WRITE_NUM_NULL_FILL</td>
<td>Defines the numeric or hexadecimal value that PowerExchange uses to replace null values in an unpacked numeric field.</td>
<td>“NRDB_WRITE_NUM_NULL_FILL Statement” on page 122</td>
</tr>
<tr>
<td>TEXT_EOF_FOR_BINARY</td>
<td>Controls whether PowerExchange ignores hexadecimal value 1A when reading binary flat files on Linux, UNIX, or Windows.</td>
<td>“TEXT_EOF_FOR_BINARY Statement” on page 152</td>
</tr>
</tbody>
</table>

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:

<table>
<thead>
<tr>
<th>Statement</th>
<th>Description</th>
<th>Reference</th>
</tr>
</thead>
<tbody>
<tr>
<td>CAPI_CONNECTION - ORAD</td>
<td>Specifies the CAPI parameters to use for PowerExchange Express CDC for Oracle sources.</td>
<td>“CAPI_CONNECTION - ORAD Statement” on page 56</td>
</tr>
<tr>
<td>CAPI_CONNECTION - ORCL</td>
<td>Specifies the CAPI parameters for PowerExchange Oracle CDC with LogMiner sources.</td>
<td>“CAPI_CONNECTION - ORCL Statement” on page 58</td>
</tr>
<tr>
<td>ORA_ERRORFILE</td>
<td>Specifies the name of the user-customized SQL error file that PowerExchange uses for Oracle bulk data movement operations.</td>
<td>“ORA_ERRORFILE Statement” on page 123</td>
</tr>
<tr>
<td>ORACLE_CAPTURE_TYPE</td>
<td>Specifies which PowerExchange Oracle CDC solution is in use for a PowerExchange installation: PowerExchange Express CDC for Oracle or PowerExchange Oracle CDC with LogMiner.</td>
<td>“ORACLE_CAPTURE_TYPE Statement” on page 124</td>
</tr>
</tbody>
</table>
### ORACLE_UNHANDLED_NUMASCHAR

Controls whether PowerExchange handles the following types of numeric columns in Oracle source tables in a manner that prevents data loss:
- NUMBER columns that have a precision greater than 28 or an undefined length
- FLOAT columns that have a precision greater than 15 significant digits

### 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.

### ORACLEID

Specifies the Oracle source database and connection information for PowerExchange CDC with Oracle LogMiner.

### 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:

<table>
<thead>
<tr>
<th>Statement</th>
<th>Description</th>
<th>Reference</th>
</tr>
</thead>
<tbody>
<tr>
<td>VSAM</td>
<td>Specifies the number of buffers that PowerExchange uses for data and index control intervals when processing VSAM data sets.</td>
<td>“VSAM Statement” on page 161</td>
</tr>
</tbody>
</table>

### 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:

<table>
<thead>
<tr>
<th>Statement</th>
<th>Description</th>
<th>Reference</th>
</tr>
</thead>
<tbody>
<tr>
<td>AS400EVENTMSGQ</td>
<td>Specifies an event message queue on an i5/OS system to which PowerExchange writes messages.</td>
<td>“AS400EVENTMSGQ Statement” on page 41</td>
</tr>
<tr>
<td>AS400USRJRNCODE</td>
<td>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.</td>
<td>“AS400USRJRNCODE Statement” on page 42</td>
</tr>
<tr>
<td>CONSOLE_TRACE</td>
<td>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.</td>
<td>“CONSOLE_TRACE Statement” on page 78</td>
</tr>
<tr>
<td>CPX_DIR</td>
<td>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.</td>
<td>“CPX_DIR Statement” on page 80</td>
</tr>
<tr>
<td>RMTRDBDIRE</td>
<td>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.</td>
<td>“RMTRDBDIRE Statement” on page 134</td>
</tr>
</tbody>
</table>

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:

<table>
<thead>
<tr>
<th>Statement</th>
<th>Description</th>
<th>Reference</th>
</tr>
</thead>
<tbody>
<tr>
<td>CAPT_PATH</td>
<td>Specifies the path to the local directory on a Linux, UNIX, and Windows system that contains the control files for CDC.</td>
<td>“CAPT_PATH Statement” on page 74</td>
</tr>
<tr>
<td>CAPT_XTRA</td>
<td>Specifies the path to the local directory on a Linux, UNIX, and Windows system that stores extraction maps for CDC.</td>
<td>“CAPT_XTRA Statement” on page 75</td>
</tr>
<tr>
<td>CMDNODE</td>
<td>Linux, UNIX, and Windows. Specifies connection information for a PowerExchange process to which you want to send pwxcmd commands.</td>
<td>“CMDNODE Statement” on page 75</td>
</tr>
<tr>
<td>Statement</td>
<td>Description</td>
<td>Reference</td>
</tr>
<tr>
<td>---------------------------</td>
<td>-------------------------------------------------------------------------------------------------------</td>
<td>------------------------------------------</td>
</tr>
<tr>
<td>CONSOLE_CODEPAGE</td>
<td>Linux, UNIX, and Windows. Specifies the code page that PowerExchange uses to display message text on consoles.</td>
<td>&quot;CONSOLE_CODEPAGE Statement&quot; on page 78</td>
</tr>
<tr>
<td>LOG_CODEPAGE</td>
<td>Windows. Specifies the code page that PowerExchange uses to write messages to the log file.</td>
<td>&quot;LOG_CODEPAGE Statement&quot; on page 111</td>
</tr>
<tr>
<td>LOGPATH</td>
<td>Linux, UNIX, and Windows. Defines the directory location where PowerExchange writes message log files on Linux, UNIX, and Windows systems.</td>
<td>&quot;LOGPATH Statement&quot; on page 112</td>
</tr>
<tr>
<td>TEXT_EOF_FOR_BINARY</td>
<td>Linux, UNIX, and Windows. Controls whether PowerExchange ignores hexadecimal value 1A when reading binary flat files on Linux, UNIX, or Windows.</td>
<td>&quot;TEXT_EOF_FOR_BINARY Statement&quot; on page 152</td>
</tr>
</tbody>
</table>

### 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:

<table>
<thead>
<tr>
<th>Statement</th>
<th>Description</th>
<th>Reference</th>
</tr>
</thead>
<tbody>
<tr>
<td>ABEND_SW</td>
<td>Controls whether PowerExchange activates additional error routines to handle abends.</td>
<td>&quot;ABEND_SW Statement&quot; on page 34</td>
</tr>
<tr>
<td>BS</td>
<td>Specifies the block size that PowerExchange uses when dynamically allocating target data sets on z/OS if you do not provide a block size.</td>
<td>&quot;BS Statement&quot; on page 43</td>
</tr>
<tr>
<td>CAPI_CONNECTION - LRAP</td>
<td>Specifies the CAPI parameters needed for the Log Read API (LRAPI) component of the PowerExchange Logger for MVS.</td>
<td>&quot;CAPI_CONNECTION - LRAP Statement&quot; on page 51</td>
</tr>
<tr>
<td>CONSOLE_TRACE</td>
<td>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.</td>
<td>&quot;CONSOLE_TRACE Statement&quot; on page 78</td>
</tr>
<tr>
<td>DATAMAP_SERVER</td>
<td>Specifies the node name of the PowerExchange Listener that acts as the server for access requests to the file that stores data maps.</td>
<td>&quot;DATAMAP_SERVER Statement&quot; on page 81</td>
</tr>
<tr>
<td>DISP</td>
<td>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.</td>
<td>&quot;DISP Statement&quot; on page 91</td>
</tr>
<tr>
<td>Statement</td>
<td>Description</td>
<td>Reference</td>
</tr>
<tr>
<td>--------------------</td>
<td>-----------------------------------------------------------------------------</td>
<td>----------------------------------------</td>
</tr>
<tr>
<td>DM RESOURCE</td>
<td>Specifies the suffix that PowerExchange uses to create the PowerExchange-specific resource profile that controls access to the DATAMAPS data set.</td>
<td><a href="#">“DM RESOURCE Statement” on page 92</a></td>
</tr>
<tr>
<td>DM_SUBTASK</td>
<td>Controls how the PowerExchange Listener accesses the file that stores data maps.</td>
<td><a href="#">“DM_SUBTASK Statement” on page 93</a></td>
</tr>
<tr>
<td>DMXCACHE_DELETEECSA</td>
<td>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.</td>
<td><a href="#">“DMXCACHE_DELETEECSA Statement” on page 94</a></td>
</tr>
<tr>
<td>DMXCACHE_MAX_MEMORY_MB</td>
<td>Enables PowerExchange data maps caching and configures the maximum size for the data maps cache in megabytes.</td>
<td><a href="#">“DMXCACHE_MAX_MEMORY_MB Statement” on page 94</a></td>
</tr>
<tr>
<td>DMXCACHE_MULTIPLEJOBS</td>
<td>Determines whether PowerExchange runs data maps caching in multiple-jobs or single-job mode.</td>
<td><a href="#">“DMXCACHE_MULTIPLEJOBS Statement” on page 95</a></td>
</tr>
<tr>
<td>ENQMAJORNAME</td>
<td>Specifies the major name for ENQ macros that PowerExchange issues to lock PowerExchange data sets and resources.</td>
<td><a href="#">“ENQMAJORNAME Statement” on page 99</a></td>
</tr>
<tr>
<td>ENQSYSTEMS</td>
<td>Controls whether PowerExchange uses a scope of SYSTEM or SYSTEMS for ENQ macros that it issues to lock PowerExchange data sets and resources.</td>
<td><a href="#">“ENQSYSTEMS Statement” on page 99</a></td>
</tr>
<tr>
<td>GDGLOCATE</td>
<td>Controls whether PowerExchange instructs dynamic allocation to use a LOCATE to get the most recent catalog information for generation data set allocation requests.</td>
<td><a href="#">“GDGLOCATE Statement” on page 101</a></td>
</tr>
<tr>
<td>JOBCLASS</td>
<td>Specifies the job class that PowerExchange uses for CA IDMS/DB metadata and DB2 LOAD utility batch jobs.</td>
<td><a href="#">“JOBCLASS Statement” on page 108</a></td>
</tr>
<tr>
<td>LRECL</td>
<td>Specifies the logical record length that PowerExchange uses when dynamically allocating target data sets on z/OS, if you do not provide LRECL information.</td>
<td><a href="#">“LRECL Statement” on page 114</a></td>
</tr>
<tr>
<td>MVSDB2AF</td>
<td>Specifies which DB2 for z/OS attachment facility PowerExchange uses for DB2 bulk data movement operations.</td>
<td><a href="#">“MVSDB2AF Statement” on page 116</a></td>
</tr>
<tr>
<td>NETPORT</td>
<td>Defines information about netport jobs and associates the netport job with a specific listener port.</td>
<td><a href="#">“NETPORT Statement” on page 117</a></td>
</tr>
<tr>
<td>PC_AUTH</td>
<td>Controls whether the PowerExchange Listener uses its MVS Program Call (PC) services routine to acquire the authorization to access CA IDMS/DB.</td>
<td><a href="#">“PC_AUTH Statement” on page 130</a></td>
</tr>
<tr>
<td>Statement</td>
<td>Description</td>
<td>Reference</td>
</tr>
<tr>
<td>------------</td>
<td>-----------------------------------------------------------------------------</td>
<td>-----------------------------------------------</td>
</tr>
<tr>
<td>RACF_CLASS</td>
<td>Specifies the RACF class name that PowerExchange uses when checking PowerExchange-specific resource profiles.</td>
<td>“RACF_CLASS Statement” on page 132</td>
</tr>
<tr>
<td>RECFM</td>
<td>Specifies the record format that PowerExchange uses when dynamically allocating target data sets on z/OS, if you do not provide RECFM information.</td>
<td>“RECFM Statement” on page 133</td>
</tr>
<tr>
<td>RELEASE</td>
<td>Controls whether PowerExchange requests or does not request to release unused space for data sets that it dynamically allocates.</td>
<td>“RELEASE Statement” on page 134</td>
</tr>
<tr>
<td>SPACE</td>
<td>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.</td>
<td>“SPACE Statement” on page 139</td>
</tr>
<tr>
<td>STATS</td>
<td>Controls whether PowerExchange writes SMF statistics records for the PowerExchange Listener to SMF or to a file.</td>
<td>“STATS Statement” on page 144</td>
</tr>
<tr>
<td>SUBMITTIMEOUT</td>
<td>Specifies the time, in seconds, that a PowerExchange Listener waits to receive notification from a spawned batch job that it has started.</td>
<td>“SUBMITTIMEOUT Statement” on page 145</td>
</tr>
<tr>
<td>SUP_FUNC</td>
<td>Defines the function name of the zIIP enclave for the PowerExchange Listener to the IBM Workload Manager.</td>
<td>“SUP_FUNC Statement” on page 146</td>
</tr>
<tr>
<td>SUP_REQUEST</td>
<td>Defines the name of the requestor for zIIP facilities.</td>
<td>“SUP_REQUEST Statement” on page 146</td>
</tr>
<tr>
<td>SUP_SSNAME</td>
<td>Defines the subsystem name that identifies the PowerExchange Listener started task to the IBM Workload Manager when zIIP exploitation is enabled.</td>
<td>“SUP_SSNAME Statement” on page 146</td>
</tr>
<tr>
<td>SUP_SSTYPE</td>
<td>Defines the name that the IBM Workload Manager uses as the subsystem type for the enclave for zIIP.</td>
<td>“SUP_SSTYPE Statement” on page 147</td>
</tr>
<tr>
<td>TAPEWAIT</td>
<td>Controls whether netport jobs wait or do not wait for tape volumes.</td>
<td>“TAPEWAIT Statement” on page 149</td>
</tr>
<tr>
<td>UNIT</td>
<td>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.</td>
<td>“UNIT Statement” on page 159</td>
</tr>
<tr>
<td>USESUP</td>
<td>Controls whether PowerExchange uses zIIP for zIIP-enabled PowerExchange Listener functions.</td>
<td>“USESUP Statement” on page 160</td>
</tr>
</tbody>
</table>
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:

<table>
<thead>
<tr>
<th>Notation</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>italic</td>
<td>Italic indicates variables that you must replace with a name or value. For example: ADABAS_DEFAULT_DBID=dbid</td>
</tr>
<tr>
<td>{}</td>
<td>Single braces enclose alternative entries. Use only one of the entries. Do not type the braces when you enter the option. For example: ABEND_SW={N</td>
</tr>
<tr>
<td>[]</td>
<td>Brackets surround optional parameters for a statement. Do not type the brackets when you enter the parameter. For example: DB2ID=(db2_subsystem[,plan][,module_name])</td>
</tr>
<tr>
<td></td>
<td>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: CREDENTIALS_CASE={A</td>
</tr>
<tr>
<td>_</td>
<td>Underline indicates the default used if you do not specify any of the alternatives. For example: ABEND_SW={N</td>
</tr>
<tr>
<td>,</td>
<td>A comma separates the parameters on a statement. For example: DB2ID=(db2_subsystem[,plan][,module_name])</td>
</tr>
<tr>
<td>...</td>
<td>An ellipsis indicates additional parameters that do not appear in the syntax. For example: TYPE=(AS4J, ...)</td>
</tr>
</tbody>
</table>

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:** UNIX, Windows, and 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 and you are running one of the following Adabas versions:

- On z/OS, Adabas version 7 or later
- On UNIX or Windows, Adabas version 5 or later

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.

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: UNIX, Windows, and z/OS

Data Sources: Adabas

Required: No

Syntax:

\[
\text{ADABAS_DEFAULT_DBID}=\{\text{dbid}\} \mid 0
\]

Value: For the \text{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_PREFIX Statement

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

Operating Systems: UNIX, Windows, and 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: UNIX, Windows, and 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:** UNIX, Windows, and 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. The following Adabas versions provide this support:

- On z/OS, Adabas version 7 or later
- On UNIX or Windows, Adabas version 5 or later

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: UNIX, Windows, and 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: UNIX, Windows, and 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.

ADAUSER 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 APPBUFSIZEEDYN statement in the DBMOVER configuration file.

**Operating Systems:** All

**Related Statements:** APPBUFSIZEEDYN

**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.

**APPBUFSIZEEDYN Statement**

The APPBUFSIZEEDYN 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 APPBUFSIZEEDYN=Y, PowerExchange resizes the application buffers for individual connection as needed.

The APPBUFSIZEEDYN 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:**

```
APPBUFSIZEEDYN={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 an event message queue on an i5/OS system to which PowerExchange writes messages.

**Operating Systems:** i5/OS

**Data Sources:** DB2 for i5/OS

**Related Statements:** CAPI_CONNECTION - ASJ

**Required:** No
Syntax:

\texttt{AS400EVENTMSGQ=library/queue\_name}

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

- \texttt{library}
  - The name of the library that contains the message queue.

- \texttt{queue\_name}
  - The message queue name.

**Usage Notes:** PowerExchange writes messages to the specified message queue when ALWCLRPFM=Y, ALWLBRFSH=Y, or AS4JRNEYEXIT=Y is specified on the AS4J CAPI\_CONNECTION statement, as follows:

- If ALWCLRPFM=Y, PowerExchange writes the DTL3002 message to the queue if PowerExchange encounters a journal entry for a CLRPFM command. PowerExchange cannot replicate changes that a CLRPFM command makes.
- If ALWLBRFSH=Y, PowerExchange writes message DTL3003I for an *ADDLIB (add library) event and message DTL3004I for a *DTLLIB (delete library) event to the 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 AS4JRNEYEXIT=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 event message queues, 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 ALWLBRFSH=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:**

\texttt{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.

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 pwxccl.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 DTLCONNOVR SQL escape sequence override.

**Related Topics:**

- “Consumer API (CAPI) Connection Statements” on page 166

**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 i5/OS CDC sources.

**Operating Systems:** i5/OS

**Data Sources:** DB2 for i5/OS

**Related Statements:** CAPI_CONNECTION - UOWC, AS400EVENTMSGQ, AS400USRJRNCODE

**Required:** Yes for DB2 for i5/OS CDC

Syntax:

```
CAPI_CONNECTION=([DLLTRACE=trace_id]
,NAME=capi_connection_name
,TRACE=trace_name]
,TYPE=(AS4J
 [,ALWCLRFPF={N|Y}]
 [,ALWLIBRFSH={N|Y}]
 [,ALWNONREN={N|Y}]
 [,ALWPARTIAL={N|Y}]
 [,AS4JRNEXIT={N|Y}]
 [,EOF={N|Y}]
```

---

**Chapter 2: DBMOVER Configuration File**
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 i5/OS CDC processing stops or continues when PowerExchange encounters changes that result from an i5/OS Clear Physical File Member (CLRPFM) command issued against a DB2 table registered for capture. PowerExchange cannot capture changes that result from a CLRPFM command.

Enter one of the following options:

- **N.** PowerExchange CDC processing stops when PowerExchange detects changes from a CLRPFM command.
- **Y.** PowerExchange ignores the CLRPFM command 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.

**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.

Default is N.

**ALWLBRFSH={N|Y}**

Optional. Controls whether you can refresh the CDC interest list after you add or delete i5/OS 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 i5/OS does not provide an NR entry, CDC processing ends abnormally.

- **Y.** Generate an NR entry so that CDC processing can continue when i5/OS 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 i5/OS system. PowerExchange records the journal receivers that it is processing in a lock file called PWXJRNLCKP 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}**

Optional. Controls the type of value that PowerExchange uses to populate the DTL__CAPXUSER column of each change record.

Enter one of the following options:

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

  **Note:** If you set ALWLIBRFSH to Y to refresh the CDC library interest list for library additions or deletions, 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.** Uses the library name, file name, and data member name of the file where the change was made.

Default is N.

**NOCCUOWSZ=uow_size**

Optional. Size, in number of records, of the UOW that PowerExchange creates when reading change records from i5/OS 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 i5/OS 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**:

```plaintext
CAPI_CONNECTION=[
   [DLLTRACE=trace_id]
   ,NAME=capi_connection_name
   [,TRACE=trace_name]
   ,TYPE=(CAPX
        ,DFLTINST=instance_name
        [,FILEWAIT={seconds}]])
   [,NOSEQVAL={Y|N}]
   [,RSTRADV={seconds}]
   [,RSTRANMODE={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 i5/OS, DB2 for z/OS, Oracle, and VSAM, the name that is displayed in the Instance field for the registration group.
- 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.
- For Datacom, the name of the Multi-User Facility (MUF) in the MUF Name field.
- For a DB2 for Linux, UNIX, and Windows source, the name of the database in the Database field.
- For an IDMS log-based source, the name of the database in the DB Name field.
- For an IMS source, the recon identifier for the database in the RECON ID field.

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}

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

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.

For the seconds variable, enter a number from 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.

For DB2 for 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 because PowerExchange returns an empty UOW after each UOW processed.

RSTRANMODE={N|Y}

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.

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 MVS.

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
 [,EOF=NO|Y])
 ,LOG=logger_id
 [,UIDFMT=(ALL|CONN|CORR|CTYPE|PLAN|UID)])
 )
```

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.

EOF={N|Y}
   Optional. Controls whether PowerExchange stops change data extractions after reaching the end-of-log (EOL).
   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.
   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.

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.

UIDFMT={ALL|CONN|CORR|CTYPE|PLAN|UID}
   Optional. For DB2 for z/OS data sources, controls the data that PowerExchange returns in the DTL__CAPXUSER field.
Enter one of the following options:

- **ALL.** Requests the information for all of the other options. PowerExchange provides this information in a colon-delimited list in the following format:
  
  ```
  UID:PLAN:CORR:CONN:CTYPE
  ```

- **CONN.** DB2 connection identifier when the change was made.
- **CORR.** DB2 correlation identifier when the change was made.
- **CTYPE.** DB2 connection type when the change was made.
- **PLAN.** DB2 plan name used when the change was made.
- **UID.** User ID that made the change.

Default is UID.

**Restriction:** You can specify only one option. If you need more than one option, enter **ALL.**

### 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:**

```c
CAPI_CONNECTION=(
  [DLLTRACE=trace_id]
  ,NAME=capi_connection_name
  [,TRACE=trace_name]
  ,TYPE=(MSQL
    [,DISTDB=distribution_database
      ,DISTSRV=distribution_server
      ],BATCHSIZE=number]
    [,DWFLAGS={flag1flag2flag3|NNN}]
    [,EOF={N|Y}]
    [,MULTIPUB={N|Y}]
    [,POLWAIT={seconds|1}]
    [,RSTRADV=seconds]
    [,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=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=(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.

**Important:** This name is different from the network name of the SQL Server instance if the distribution database resides on a different server.

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 PowerExchanges issues the data read query more often.

DWFLAGS={flag1flag2flag3|NNN}  
Optional. Series of three positional parameters that control whether processing stops or continues when data loss, truncation, or schema changes 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 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 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 N to stop processing.

Default is NNN.

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 pwxccl.cfg configuration file.
- For CDC sessions that use ODBC connections, enter 0 for the WAITTIME parameter in the ODBC data source.

**MULTIPUB={NIY}**

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.

For the seconds variable, enter a number from 1 through 2147483647. Default is 1.

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.

For the seconds variable, enter a number from 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.

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 - 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:

```plaintext
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 `pwx_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.

- You cannot define both ORAD CAPI_CONNECTION and ORCL CAPI_CONNECTION statements in the same dbmover.cfg file. If you use PowerExchange Express CDC for Oracle, define ORAD CAPI_CONNECTION statements. If you use PowerExchange Oracle CDC with LogMiner, define ORCL CAPI_CONNECTION statements. If you need to run both PowerExchange Express CDC for Oracle and PowerExchange Oracle CDC with LogMiner sessions against the same Oracle instance, use separate dbmover.cfg files, PowerExchange Listeners, and PowerExchange Loggers.

- 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 PowerCenter CDC session

**CAPI_CONNECTION - ORCL Statement**

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

**Operating Systems:** Linux, UNIX, and Windows

**Data Sources:** Oracle

**Related Statements:** CAPI_CONNECTION - UOWC and ORACLEID
Required: Yes for PowerExchange Oracle CDC with LogMiner

Syntax:

CAPI_CONNECTION=(\[DLLTRACE=trace_id\],NAME=capi_connection_name
\[,TRACE=trace_name\]
\[,TYPE=(ORCL, ... )\]
\[,ARRAYSIZE=array_size|100]\n\[,BYPASSUF={Y|N}]\n\[,CATBEGIN=hh:mm[00:00]\n\[,CATEND=hh:mm[24:00]\n\[,CATINT=minutes|1440]\n\[,COMMITINT=minutes|2]\n\[,GENRLOCK={N|Y}]\n\[,IGNUFMSG={N|Y}]\n\[,LOGDEST=logdest_id]\n\[,LTHREAD=instance_number]\n\[,ONLINECAT={N|Y}]\n\[,ORACOLL=collection_id]\n\[,ROWID={N|Y|ALLOW}]\n\[,SELMTRY=retry_number|1000]\n\[,SNGLINST={N|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=(ORCL, ... )

Required. Type of CAPI_CONNECTION statement. For PowerExchange Oracle CDC with LogMiner sources, this value must be ORCL.

ARRAYSIZE=(array_size|100)

Optional. The number of rows of the prefetch array that PowerExchange uses to read the Oracle redo logs.

Enter a number from 0 through 2147483647. Default is 100.

Note: A value less than 100 can degrade Oracle CDC with LogMiner performance. A value of 0 disables prefetch. Enter 0 only at the direction of Informatica Global Customer Support.

BYPASSUF={N|Y}

Optional. Controls whether PowerExchange ends abnormally or issues a warning message when Oracle LogMiner returns an unformatted log record.

LogMiner returns unformatted log records when Global Temporary Tables are updated, or when ONLINECAT=Y and the log data that is being read is inconsistent with the catalog.
Enter one of the following options:

- **N.** PowerExchange ends with an error whenever it receives an unformatted log record from Oracle LogMiner.
- **Y.** PowerExchange writes a warning message to the PowerExchange message log that indicates unformatted log data was found and then continues processing. Depending on the amount of unformatted log data, PowerExchange might write many warning messages. To suppress these warning messages, specify Y for the IGNUFMSG parameter.

Default is **N**.

**Tip:** Enter Y if the Oracle instance contains Global Temporary tables. Otherwise, do not include the BYPASSUF parameter.

**CATBEGIN=〈hh:mm[00:00]〉**

Optional. Earliest time of day, in a 24-hour clock format, at which PowerExchange requests Oracle to write a copy of the Oracle catalog to the redo logs.

If you specify a value for the CATBEGIN parameter, you must also specify a value for the CATEND parameter.

Default is 00:00.

**CATEND=〈hh:mm[24:00]〉**

Optional. Latest time of day, in a 24-hour clock format, at which PowerExchange requests Oracle to write a copy of the Oracle catalog to the redo logs.

If you specify a value for the CATEND parameter, you must also specify a value for the CATBEGIN parameter.

Default is 24:00.

**CATINT={minutes[1440]}**

Optional. Time interval, in minutes, between requests to copy the Oracle catalog to the redo logs.

Enter a number from 1 through 1440. Default is 1440.

If this interval elapses but the time is outside the time period that is specified by the CATBEGIN and CATEND parameters, PowerExchange does not request Oracle to make a copy of the Oracle catalog. Instead, PowerExchange waits the amount of time that you specify in the CATBEGIN parameter to request a catalog copy.

**COMMITINT={minutes[5]}**

Optional. Time interval, in minutes, between the SQL COMMIT operations that PowerExchange issues to commit the transactions that the Oracle LogMiner session generates.

Enter a number from 1 through 60. Default is 5.

Although PowerExchange does not update data in user tables while reading change data from the redo logs, the Oracle LogMiner interface automatically generates transactions for the LogMiner sessions that PowerExchange initiates. Oracle leaves these transactions open, or in-flight, until the LogMiner session ends.

To be able to restart change data extraction operations efficiently, PowerExchange must occasionally issue SQL COMMIT operations to end these in-flight transactions. Otherwise, the restart of all future real-time extraction operations might be impacted because PowerExchange always begins reading change data at the beginning of the oldest in-flight UOW.
GENRLOCK={N|Y}
Optional. Controls whether PowerExchange generates a safe restart point for requests for restart points that match the current end-of-log (EOL).

Enter one of the following options:

- **N.** PowerExchange generates restart points that match the current EOL, ignoring any in-flight transactions for the source tables.
- **Y.** PowerExchange generates safe restart points for source tables.

Default is **N.**

A safe restart point for a source table is a point in the change stream that does not skip any in-flight UOWs for that table. To generate a safe restart point for a source table, PowerExchange gets an exclusive lock on the table to stop further changes. PowerExchange then searches the Oracle catalog for the point in the change stream that matches the earliest active transaction for the table and uses this point as the restart point. If no in-flight UOWs exist for a table, PowerExchange uses the current EOL. PowerExchange releases the lock on the source table after restart point generation completes. Then changes can be written to the table again.

PowerExchange generates restart tokens that match the current EOL in the following situations:

- You cold start the PowerExchange Logger for Linux, UNIX, and Windows and the pwxccl configuration file does not specify the SEQUENCE_TOKEN and RESTART_TOKEN parameters. PowerExchange gets locks for all of the tables that are associated with active capture registrations to be used for PowerExchange Logger processing.
- The restart token file for a CDC session specifies the CURRENT_RESTART option on the RESTART1 and RESTART2 special override statements. PowerExchange gets locks only for the tables in the CDC session to which the special override statements apply.
- A database row test in the PowerExchange Navigator that uses the SELECT CURRENT_RESTART SQL statement. PowerExchange gets a lock for the table represented by capture registration associated with the extraction map used in the database row test.
- A DTLUAPPL utility operation that uses the RSTTKN GENERATE option. PowerExchange gets a lock for the table represented by the capture registration specified in the utility control statements.

IGNUFSMSG={N|Y}
Optional. Controls whether PowerExchange writes warning messages to the PowerExchange message log file for unformatted data records.

Enter one of the following options:

- **N.** PowerExchange does not write any warning messages.
- **Y.** PowerExchange writes warning messages.

Default is **N.**

LOGDEST=\logdest_id
Optional. The numeric identifier for the archive log destination that you want to force PowerExchange to use. This archive log destination must be local to the Oracle instance that PowerExchange is using.
Enter a number from 1 through 10.

For example, to use archived logs from the destination set by the LOG_ARCHIVE_DEST_3 parameter in the init.ora file, specify LOGDEST=3.

The SNGINST parameter affects how PowerExchange uses the archive log destination and the Oracle instance that you specify in LOGDEST and LGTHREAD.

If you specify Y for the ONLINECAT parameter, PowerExchange validates and then ignores the LOGDEST and LGTHREAD parameters.

**LGTHREAD=instance_number**

Optional. The instance number for the Oracle instance. PowerExchange uses this instance number to identify the archived redo logs to process.

Enter a number from 1 through 2147483647.

For non-RAC environments, if you specify this parameter, set it to 1.

The SNGINST parameter affects how PowerExchange uses the archive log destination and the Oracle instance that you specify in LOGDEST and LGTHREAD.

If you specify Y for the ONLINECAT parameter, PowerExchange validates and then ignores the LOGDEST and LGTHREAD parameters.

**ONLINECAT=(N|Y)**

Optional. Controls whether PowerExchange directs Oracle LogMiner to use the Oracle online catalog or the copy of the catalog in the redo logs to format log data for CDC.

Enter one of the following options:

- **N.** Oracle LogMiner uses the copy of the catalog from the archived redo logs and PowerExchange tracks schema changes to ensure that data loss does not occur.
- **Y.** Oracle LogMiner uses the online catalog and PowerExchange cannot track schema changes.

Default is N.

When you configure PowerExchange to use the online catalog for formatting log data, PowerExchange still uses catalog copies to determine the restart point for change data extraction operations. Therefore, you must copy the online catalog to the Oracle redo logs on a regular basis.

Change data extraction operations generally initialize faster when you configure PowerExchange to create LogMiner sessions with the online catalog instead of a catalog copy. However, when LogMiner uses the online catalog, it does not track DDL changes, and cannot format log records for tables that have schema changes.

If LogMiner uses the online catalog and you make schema changes while LogMiner is reading log data, LogMiner passes unformatted log records for subsequent changes to PowerExchange. If you specify N for the BYPASSUF parameter or accept the default value of N, PowerExchange fails the extraction request after Oracle passes the first unformatted record. Otherwise, PowerExchange skips the unformatted record and continues processing, which results in change data loss.

Therefore, specify N for the ONLINECAT parameter, or allow it to default, if you have the following requirements:

- You specify Y for the BYPASSUF parameter and need to change the schema of tables registered for capture while change data extraction operations are running.
- You need to start an extraction from a point in the Oracle redo logs that contains table data that PowerExchange captured under a previous schema.
ORACOLL=collection_id

Required. Oracle collection identifier, which must match the value specified in the ORACLEID statement.

ROWID=\{N|Y|ALLOW\}

Controls whether Oracle physical rowid values are included in captured change records for tables that do not have Oracle row movement enabled. PowerExchange writes the rowid values to the PowerExchange-generated DTL__CAPXROWID column. For example, use this parameter if you have unkeyed source tables on which you need to perform some processing that requires a unique row ID when extraction sessions run.

Enter one of the following options:

- N. Do not capture rowid values. The DTL__CAPXROWID column contains null values.
- Y. Capture rowid values for tables that do not have row movement enabled and write the values to the DTL__CAPXROWID column in change records. If a table has row movement enabled, capture processing ends abnormally.
- ALLOW. Capture rowid values for tables that do not have row movement enabled and write the values to the DTL__CAPXROWID column in change records. If a table has row movement enabled, return null values to the DTL__CAPXROWID column and continue capture processing. You might want capture processing to continue if you do not need rowid values for the tables that have row movement enabled.

Note: This parameter pertains to PowerExchange Oracle CDC with LogMiner. If you use PowerExchange Express CDC for Oracle, set the similar ROWID parameter in the OPTIONS statement of the Express CDC configuration file instead.

Default is N.

SELRETRY=\{retry_number|1000\}

Optional. The number of times that PowerExchange immediately loops back to the Oracle LogMiner call before implementing a graduated-scale wait loop.

After PowerExchange retries the call to LogMiner the specified number of times, PowerExchange implements a wait interval between each subsequent retry. The wait interval begins at one millisecond and gradually increases to one second. When LogMiner returns data, PowerExchange resets the wait interval to 0, and the process begins again for the next call to LogMiner.

For the retry_number variable, enter a number from 0 through 2147483647. Default is 1000.

If you specify a nonzero value, PowerExchange uses nonblocking SQL to ensure that it can process a user request to shut down an extraction session in a timely manner.

If you specify 0, PowerExchange does not use nonblocking SQL. This setting improves CPU consumption but can prolong shutdown of an extraction session. On quiescent Oracle instances, PowerExchange does not honor a shutdown request until Oracle returns log data. On Oracle instances where update activity is occurring, shutdown behavior does not noticeably change.

Important: If you capture change data from an Oracle 12.1.0.1 or later source, set the SELRETRY parameter to 0. Otherwise, the Oracle LogMiner sessions for PowerExchange CDC fail when trying to fetch change data.

SNGLINST=\{N|Y\}

Optional. Controls whether PowerExchange uses only the archived redo logs from a specific Oracle instance and archive log destination.
Enter one of the following options:

- N. PowerExchange uses the specified Oracle instance to search for the archived redo logs that contain copies of the Oracle catalog. After PowerExchange passes these logs to an Oracle LogMiner session, LogMiner determines the other archived redo logs to read.
- Y. PowerExchange uses only the archive log destination and Oracle instance that you specify in LOGDEST and LGTHREAD parameters to read archived redo logs. LogMiner does not read any other archived redo logs. After PowerExchange processes the logs from the specified location, the change data extraction operation ends.

If you specify Y, you must also specify the LOGDEST and LGTHREAD parameters to identify the archive log destination and Oracle instance to use. In a RAC environment, you must run separate change data extraction processes for all remaining Oracle instances in the RAC and determine how to properly merge the change data so that it can be applied to the targets.

Default is N.

**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
 [,CCATALOG={capture_catalog|creator.DTLCATALOG}]
 [,DBCNN=database_name]
 [,EPWD=encrypted_password]
 [,LARGEOPS=number_of_operations]
 [,MEMCACHE={cache_size|1024}]
 [,MONITORINT={minutes|2}]
 [,PASSWORD=password]
 [,RSTRADV=seconds]
 [,SPACEPRI={AUTO|MAX|NONE|nn}]
 [,THREADING={AUTO|MAX|NONE|nn}]
 [,UDBSHEMA={schema}]
 [,UPDINT={seconds|600}]
 [,UPDREC={records|1000}]
 [,USERID=user_id]
}
```

**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=(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.
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 CDC sources, Oracle CDC with LogMiner sources, and z/OS CDC sources

Related Statements: CAPI_CONNECTION - AS4J, CAPI_CONNECTION - LRAP, and CAPI_CONNECTION - ORCL

Required: Yes, for CDC for the specified sources

Syntax:

```
CAPI_CONNECTION=<([DLC_TRACE=trace_id]
         ,NAME=capi_connection_name
         ,[TRACE=trace_name]
         ,TYPE=(UOWC
               ,CAPINAME=source_capi_name
               ,[BLKSIZE=block_size]
               ,[DATACLASS=data_class]
               ,[LIO=$number_of_operations]
               ,[MEMCACHES=cache_size[1024]]
               ,[MONITORINT={minutes[1]}]
               ,[RSTRADV={seconds}]
               ,[SPACEPRI=[primary_space[50]]
               ,[SPACETYPE=[BLK|TRK|CYL]]
               ,[SPILLKEEP=number_of_spill_files]
               ,[STORCLASS=storage_class]
               ,[TIMESTAMP=[LOG|COMMIT]])
```

**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.
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 i5/OS sources
- ORCL CAPI_CONNECTION statement for Oracle CDC with LogMiner 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:

<table>
<thead>
<tr>
<th>Data Source Type</th>
<th>Valid Values</th>
<th>Default Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>DB2 for i5/OS</td>
<td>A number from 8 through 32760</td>
<td>32760</td>
</tr>
<tr>
<td>Oracle</td>
<td>A number from 8 through 65535</td>
<td>32768</td>
</tr>
<tr>
<td>z/OS data sources</td>
<td>A number from 8 through 32760</td>
<td>18452</td>
</tr>
</tbody>
</table>

**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 SYSyyddd.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 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 349.
- **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.

**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 describes the possible values:

<table>
<thead>
<tr>
<th>Option</th>
<th>Source Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>ADA</td>
<td>Adabas sources</td>
</tr>
<tr>
<td>AS4</td>
<td>DB2 for i5/OS sources</td>
</tr>
<tr>
<td>CAPX</td>
<td>Sources for which you are extracting data in continuous extraction mode and using the PowerExchange Logger for Linux, UNIX, or Windows or PowerExchange Condense</td>
</tr>
<tr>
<td>DB2</td>
<td>DB2 for z/OS sources</td>
</tr>
<tr>
<td>DCM</td>
<td>CA Datacom sources</td>
</tr>
<tr>
<td>IDL</td>
<td>CA IDMS/DB log-based CDC sources</td>
</tr>
<tr>
<td>IML</td>
<td>IMS log-based CDC sources</td>
</tr>
<tr>
<td>IMS</td>
<td>IMS synchronous CDC sources</td>
</tr>
<tr>
<td>MSS</td>
<td>Microsoft SQL Server sources</td>
</tr>
<tr>
<td>ORA</td>
<td>Oracle sources</td>
</tr>
<tr>
<td>UDB</td>
<td>DB2 for Linux, UNIX, and Windows sources</td>
</tr>
<tr>
<td>VSAM or VSM</td>
<td>VSAM sources</td>
</tr>
</tbody>
</table>

- `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:

<table>
<thead>
<tr>
<th>CAPI_SRC_DFLT Option</th>
<th>CAPI_CONNECTION Statement Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>AS4</td>
<td>UOWC</td>
</tr>
<tr>
<td>CAPX</td>
<td>CAPX</td>
</tr>
<tr>
<td>ADA, DB2, DCM, IDL, IDM, IML, IMS, VSAM, or VSM</td>
<td>UOWC</td>
</tr>
<tr>
<td>ORA</td>
<td>UOWC for PowerExchange Oracle CDC with LogMiner ORAD for PowerExchange Express CDC for Oracle</td>
</tr>
<tr>
<td>MSS</td>
<td>MSQL</td>
</tr>
<tr>
<td>UDB</td>
<td>UDB</td>
</tr>
</tbody>
</table>

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 CAPICONN_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 pwxccl.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 DTLCCONN_OVR parameter in the odbc.ini file or the SQL escape sequence override DTLCONNOVR.

Related Topics:

- “Consumer API (CAPI) Connection Statements” on page 166

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: Microsoft SQL Server, Oracle, DB2 for Linux, UNIX, and Windows, and if you use offloading processing, other data sources on 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: Microsoft SQL Server, Oracle, and DB2 for Linux, UNIX, and Windows

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 CAPTPATH 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|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.

**{CONDENSE|LISTENER}**
- Required. PowerExchange service type.
- Enter one of the following options:
  - **CONDENSE.** PowerExchange Condense or PowerExchange Logger for Linux, UNIX, and Windows.
  - **LISTENER.** PowerExchange Listener.
- No default value is provided.

**host_name**
- Required. Host name or IP address of the target system to which you want to send commands.

**connect_port**
- Required. 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:

<table>
<thead>
<tr>
<th>Operating System</th>
<th>Default Code Page Values</th>
</tr>
</thead>
<tbody>
<tr>
<td>i5/OS</td>
<td>IBM-037,IBM-037,IBM-037</td>
</tr>
<tr>
<td>Linux, UNIX, and Windows</td>
<td>ISO-8859,ISO-8859,ISO-8859</td>
</tr>
<tr>
<td>z/OS</td>
<td>IBM-037,IBM-037,IBM-037</td>
</tr>
</tbody>
</table>

- This statement does not apply to the PowerExchange Navigator. The PowerExchange Navigator always uses the UTF-8 code page.

- If you click **AdvancedParms** 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](page 272)"

**COLON Statement**

The COLON statement defines the character that PowerExchange uses as the delimiting 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 Y.

**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_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}
```

78 Chapter 2: DBMOVER Configuration File
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[NUL]]]
 {[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[NUL]}

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

For data maps defined with data checking, the DATERANGE statement specifies the range of years that PowerExchange uses 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. For data maps defined with data checking, the year that PowerExchange uses as the starting year in the date range for DATE and TIMESTAMP fields. Default is 1800.

{end_year|2200}
  Required. For data maps defined with data checking, the year that PowerExchange uses as the ending year in the date range for DATE and TIMESTAMP fields. This value must be equal to or larger 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 completes when it detects data that is not valid in a date field. 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:**

```sql
DB2_BIN_CODEPAGE=(sbcs_ccsid, dbcs_ccsid)
```

**Parameters:**

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

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

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

```sql
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:**

```sql
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:

<table>
<thead>
<tr>
<th>Data Source</th>
<th>Sample Error File</th>
</tr>
</thead>
<tbody>
<tr>
<td>DB2 for z/OS</td>
<td>DB2ERR member in RUNLIB</td>
</tr>
<tr>
<td>DB2 for i5/OS</td>
<td>DB2ERR member in dataib/CFG</td>
</tr>
<tr>
<td>DB2 for Linux, UNIX, and Windows</td>
<td>db2err.act in the PowerExchange installation directory</td>
</tr>
</tbody>
</table>

**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:**

```plaintext
DB2CODEPAGE=(db2_subsystem

[,'DB2TRANS=[P|N|R]']
[,'MIXED=[N]']
[,'EBCDIC_CCSID={sbscs_csvid|037}]
[,'graphic_csvid|037]
[,'mixed_csvid|037}]
[,'ASCII_CCSID={sbscs_csvid|850}]
[,'graphic_csvid|65277]
[,'mixed_csvid|65534}]
[,'UNICODE_CCSID={sbscs_csvid|1200}]
[,'graphic_csvid|1200]
[,'mixed_csvid|1200}]
[,'PLAN_CCSID={sbscs_csvid|037}]
[,'graphic_csvid|037]
[,'mixed_csvid|037}]
[,'REMAPn=(current_data_csvid),(remapped_data_csid)

)
```

**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=({sbcs_ccsid|037},{graphic_ccsid|037},{mixed_ccsid|037})

Required. 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. In DB2 Version 8.1 and later, 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=**

```
{sbcs_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=**

```
{sbcs_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=**

```
{sbcs_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.

**REMAPn=**

```
(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 REMAPn 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\textsubscript{n} parameter.

You can specify up to six REMAP\textsubscript{n} parameters in a DB2CODEPAGE statement, each for a different DB2 table. Increment the \textit{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:

<table>
<thead>
<tr>
<th>DB2CODEPAGE Parameter</th>
<th>DSNTIPF Field</th>
<th>DSNHDECP Parameter</th>
</tr>
</thead>
<tbody>
<tr>
<td>EBCDIC_CCSID</td>
<td>EBCDIC CCSID</td>
<td>SCCSID (single-byte), MCCSID (mixed), GCCSID (graphic)</td>
</tr>
<tr>
<td>ASCII_CCSID</td>
<td>ASCII CCSID</td>
<td>ASCCSID (single-byte), AMCCSID (mixed) AGCCSID (graphic)</td>
</tr>
<tr>
<td>UNICODE_CCSID</td>
<td>UNICODE CCSID</td>
<td>USCCSID (single-byte), UMCCSID (mixed), UGCCSID (graphic)</td>
</tr>
<tr>
<td>MIXED</td>
<td>MIXED DATA</td>
<td>MIXED</td>
</tr>
</tbody>
</table>

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 **AdvancedParms** on the **DB2 Parameters** page of the z/OS Installation Assistant, you can enter CCSID values for the following parameters:
  - DB2CODEPAGE\_ASCII\_DBCSCCSID
  - DB2CODEPAGE\_ASCII\_MIXED\_CCSID
  - DB2CODEPAGE\_ASCII\_SBCSCCSID
  - DB2CODEPAGE\_EBCDIC\_DBCSCCSID
  - DB2CODEPAGE\_EBCDIC\_MIXED\_CCSID
  - DB2CODEPAGE\_EBCDIC\_SBCSCCSID
  - DB2CODEPAGE\_MIXED

DBMOVER Statements 87
• 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 provides multiple parameters for greater flexibility in handling code pages. For examples of how to use these parameters in various scenarios, see “DB2CODEPAGE Statement Examples” on page 278.

• 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.

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:

\[ \text{EBCDIC_CCSID}=(\text{sbcss_ccsid, graphic_ccsid, 037}) \]

**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:**

\[
\text{DB2ID} = \{ \text{db2_subsystem} \} \{, \text{plan} \} \{, \{ \text{module_name|DTLAMV8F} \} \}
\]

**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.** For DB2 9.1 and later, 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.
- If you use multiple DB2 versions on your system, specify a DB2ID statement for each subsystem that PowerExchange accesses.

**DB2PLAN Statement**

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

**Operating Systems:** z/OS
**Data Sources:** DB2 ror 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 DTLPLvrm, where \(vrm\) is the numeric value for the PowerExchange version, release, and modification level. For example, for PowerExchange 8.6.1, the \(vrm\) value is 861. For PowerExchange 9.5.0, the \(vrm\) 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:**

\[
\text{DISP}=(\text{status} \\
\quad [\text{,normal\_disp}] \\
\quad [\text{,abnormal\_disp}])
\]

**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 DMRESOURCE 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 217

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, DMRESOURCE, DMXCACHE_DELETEECSA, DMXCACHE_MAX_MEMORY_MB, DMXCACHE_MULTIPLEJOBS, RACFCLASS, 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 217

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 version of the PowerExchange messages file to use.

Data Sources: All

Required: No

Syntax:

```
DTLMSG_CODEPAGE={code_page|UTF-8}
```

Value: For the code_page variable, enter a code page identifier. Default is UTF-8.
Examples: To use the UTF-8 version of the messages file, accept the default value or define the following statement:

```
DTLM SG_CODEPAGE=UTF-8
```

PowerExchange uses the UTF-8 version of the messages file, dtmsgs.txt.

To use the localized Japanese messages file, define the following statement:

```
DTLM SG_CODEPAGE=CP943
```

PowerExchange uses the multibyte version of the messages file, dtmsg SHIFT-JIS.txt.

**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 DES and RSA Security RC2 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, with limited support on 64-bit Windows and Linux on IBM System z  
**Related Statements:** ENCRYPTLEVEL and SSL  
**Required:** No  
**Syntax:**

```
ENCRYPT={DES|N|RC2|Y}
```
Valid Values:

- **DES.** PowerExchange encrypts data by using the Digital Encryption Standard (DES) algorithm. You must specify the encryption level by coding the ENCRYPTLEVEL statement.
- **N.** PowerExchange does not encrypt data.
- **RC2.** PowerExchange encrypts data by using the RSA Security RC2 algorithm. You must specify the encryption level by coding the ENCRYPTLEVEL statement.
- **Y.** PowerExchange uses its internal proprietary encryption method to encrypt data.

Default is **N.**

Usage Notes:

- The RC2 and DES encryption types are not supported on 64-bit Windows or Linux on IBM System z.
- To ensure predictable results, specify the same values in the ENCRYPT and ENCRYPTLEVEL statements in the DBMOVER configuration files on the client and PowerExchange Listener machines.
- For individual connections, the connection attributes that you enter in the PowerCenter Workflow Manager or the Informatica Developer Tool override the values that you define for the ENCRYPT and ENCRYPTLEVEL statements in the DBMOVER configuration file on the PowerCenter Integration Service or Data Integration Service machine.

**RELATED TOPICS:**

- "PowerExchange Alternative Network Security" on page 245
- "Secure Sockets Layer Support" on page 231

**ENCRYPTLEVEL Statement**

The ENCRYPTLEVEL statement specifies the encryption level to use if you select DES or RSA Security RC2 encryption. The encryption level determines the encryption key length.

If you enter DES or RC2 in the ENCRYPT statement, you must include the ENCRYPTLEVEL statement.

**Operating Systems:** All except 64-bit Windows and Linux on IBM System z

**Related Statements:** ENCRYPT

**Required:** Yes if you use DES or RC2 encryption

**Syntax:**

```
ENCRYPTLEVEL={1|2|3}
```

**Valid Values:**

- **1.** Use a 56-bit encryption key for both DES and RC2.
- **2.** Use one of the following encryption keys:
  - For DES, a 168-bit triple DES encryption key.
  - For RC2, a 64-bit encryption key.
- **3.** Use one of the following encryption keys:
  - For DES, a 168-bit triple DES encryption key.
  - For RC2, a 128-bit encryption key.

Default is 1.
Usage Notes:

- The RC2 and DES encryption types are not supported on 64-bit Windows or Linux on IBM System z.
- To ensure predictable results, specify the same values in the ENCRYPT and ENCRYPTLEVEL statements in the DBMOVER configuration files on the client and PowerExchange Listener machines.
- For individual connections, the connection attributes that you enter in the PowerCenter Workflow Manager or the Informatica Developer Tool override the values that you define for the ENCRYPT and ENCRYPTLEVEL statements in the DBMOVER configuration file on the PowerCenter Integration Service or Data Integration Service machine.

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:

<table>
<thead>
<tr>
<th>ENQ Minor Name</th>
<th>Usage Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>DTLAMCPR</td>
<td>Capture registrations operations.</td>
</tr>
<tr>
<td>DTLACDC</td>
<td>CDCT operations.</td>
</tr>
<tr>
<td>DTLDB2</td>
<td>DB2 LOAD utility operations.</td>
</tr>
<tr>
<td>DTLILCAT</td>
<td>Updates to the log catalog for CA IDMS/DB log-based CDC.</td>
</tr>
<tr>
<td>DTLLOG</td>
<td>Updates to the standard PowerExchange message log, DTLLOG.</td>
</tr>
</tbody>
</table>

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.
**ENQMAJORNAME Statement**

**Required:** No

**Syntax:**

\[ \text{ENQMAJORNAME} = \{N|Y\} \]

**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:**

\[ \text{ERRROWNOTFOUND} = \{N|Y\} \]

**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
- USRC00

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 272

**GDGLOCATE Statement**

The GDGLOCATE controls whether PowerExchange instructs dynamic allocation to use a LOCATE to get the most recent catalog information for generation data set allocation requests.

**Operating Systems:** z/OS
**Related Statements:** NETPORT, TAPEWAIT, and WAITDSN

**Required:** No

**Syntax:**

```
GDGLOCATE=Y
```

**Value:** Enter Y. The PowerExchange Listener uses the most recent catalog information to read or write relative generations.

PowerExchange refreshes the generation table for a GDG by specifying the S99GDGNCT bit in S99FLAGS1 on dynamic allocation (SVC99) request. Dynamic allocation then uses a LOCATE to determine the relative generation number based on the most recent catalog information.

**Usage Notes:** By default, the PowerExchange Listener does not refresh the generation table after the first reference to a generation data set. After the PowerExchange Listener accesses a generation data set using a relative generation number, all future references to relative generations of the GDG access the same absolute generations that existed during the first reference.

For example, relative generation 0 for MY.GDG corresponds to data set MY.GDG.G0011V00. The first time you reference relative generation 0 through the PowerExchange Listener, PowerExchange reads absolute generation G0011V00. Then, you create a new generation, which the operating system assigns an absolute generation number of G0012V00. Afterwards, if you request relative generation 0 through the same PowerExchange Listener, PowerExchange still reads absolute generation G0011V00. In this example, if you specify Y for GDGLOCATE, PowerExchange reads absolute generation G0012V00 when you request relative generation 0 instead of G0011V00.

**Related Topics:**

- "Using Netport Jobs for Generation Data Sets" on page 177

**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.

**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**

Optional. 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)
```
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:

  \[
  \text{ICUCNVPROPERTY}=(301, \text{ASCII,ON})
  \]

- **BUFFER_INCREMENT.** PowerExchange allocates extra bytes for the buffer when converting to the code page. Enter a numeric value for this property.

  For example:

  \[
  \text{ICUCNVPROPERTY}=(301, \text{BUFFER_INCREMENT,8})
  \]

- **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:

  \[
  \text{ICUCNVPROPERTY}=(301, \text{EBCDIC,ON})
  \]

- **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:

  \[
  \text{ICUCNVPROPERTY}=(301, \text{FIX_MIN_LENGTH,ON})
  \]

- **IBM420_END_OF_WORD_SPACES.** Adds a space after certain IBM420 characters that end words.

  For example:

  \[
  \text{ICUCNVPROPERTY}=(301, \text{IBM420_END_OF_WORD_SPACES,ON})
  \]

- **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:

  \[
  \text{ICUCNVPROPERTY}=(301, \text{LOWERCASE_NONSTANDARD,ON})
  \]

- **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:

  \[
  \text{ICUCNVPROPERTY}=(301, \text{POWERCENTER_NAME,\"IBM EBCDIC Japanese\")}
  \]

- **REVERSE_EBCDIC_DIGITS.** Reverses consecutive sequences of the following bytes: F0, F1, F2, F3, F4, F5, F6, F7, F8, and F9.
For example:

\[
\text{ICUCNVPROPERTY}=(301,\text{REVERSE_EBCDIC_DIGITS,ON})
\]

- **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:

\[
\text{ICUCNVPROPERTY}=(301,\text{SQL_PARSABLE,ON})
\]

- **STATE_BYTES.** PowerExchange allocates two extra bytes for the buffer when converting to the code page.

For example:

\[
\text{ICUCNVPROPERTY}=(301,\text{STATEgetBytes,ON})
\]

- **UNKNOWN_ENDIANNESS.** Indicates that the code page has a different integer endianness, based on the system.

For example:

\[
\text{ICUCNVPROPERTY}=(301,\text{UNKNOWN_ENDIANNESS,ON})
\]

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 0xFEF4 to equivalent 0x62xx characters. Converts LAM ALEF ligature characters from 0xFEF5 through 0xFEFC to the Unicode 0x0644 character and another character.

For example:

\[
\text{ICUCNVPROPERTY}=(301,\text{UNSHAPE_ARABIC,ON})
\]

- **USE_TYPE1_FALLBACKS.** Enables type 1 fallback mappings for the specified code page.

For example:

\[
\text{ICUCNVPROPERTY}=(242,\text{USE_TYPE1_FALLBACKS,ON})
\]

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:

```plaintext
ICUCONVERTER=(CPN
     ,ADD
     ,code_page_name
     ,min_bytes
     ,max_bytes
     ,space_hex
 )
```

For the DELETE commands:

```plaintext
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:
  
  ```plaintext
  ICUCONVERTER=(301,ADD,user_showa_01,2,2,0000)
  ```

  All characters are two bytes. The hexadecimal space character is \X00\X00.

- **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:
  
  ```plaintext
  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:

  ```plaintext
  ICUALIAS=(CPN,DELETEALL)
  ```

  For example, to add the user_cp930 code page into CPN 302, include the following statement:

  ```plaintext
  ICUCONVERTER=(302,ADD,user_cp930,1,2,40)
  ```

  All characters use either one or two bytes. The hexadecimal space character is \X40.

**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 pmicud32 string appended with the 1 or b character.
   For example, the converter named ibm-937_P110-1999 is in the

IMSID Statement

The IMSID statement defines the IMS subsystem information that PowerExchange uses for 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:

```r
IMSID={ims_ssid

,dbdlib

,[,RECON={recon1

[,recon2]

[,recon3]})

}
```

Parameters:

`ims_ssid`

Required. IMS subsystem ID (SSID). If you use IMS ODBA data maps, this value must match the IMS SSID. PowerExchange uses this value to connect to the specified IMS subsystem.

If you do not use IMS ODBA data maps, PowerExchange uses the `ims_ssid` value to match the value of the registration group in the PowerExchange Navigator.

`dbdlib`

Required for IMS log-based or synchronous CDC. DBDLIB data set for DBD modules specified in IMS data maps. A valid value is a one- to eight-character alphanumeric string.

`RECON=(recon1,[recon2],[recon3])`

Optional. RECON data sets for IMS log-based CDC. If you code RECON, you must code at least one data set name. You can code 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.
- When you create an IMS data map, PowerExchange does not use the IMSID statement. However, specify an IMS SSID value in the data map that matches the IMS subsystem ID value on an IMSID statement. When you perform IMS bulk data movement operations, PowerExchange uses the IMSID statement with the IMS subsystem ID that matches the IMS SSID value in the IMS data map to determine the DBDLIB data set for the IMS database.
- For IMS log-based CDC, PowerExchange uses the IMSID statement with the IMS subsystem ID that matches the IMS SSID value in the IMS data map when you register IMS sources.

**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:

```r
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.
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 DBMOVER 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] ,TCPIP ,[port|2480] ,{send_bufsize|65536} ,{receive_bufsize|65536} ,{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. PowerExchange uses SSL authentication.

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 pwxCreatelistenerService 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 Administrator Guide.

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.
  - `DB2LDJCX`. 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.
LOG_CODEPAGE Statement

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 RECLEN parameter in the TRACING statement, PowerExchange uses the LOG_LINE_LIMIT value as the RECLEN value.
• If you enter the MONITOR parameter in the STATS statement, set the LOG_LINE_LIMIT value to 132. Otherwise, the lines in the PowerExchange Listener displaystats report might wrap awkwardly, 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:**

\[
\text{LOGPATH} = \text{directory}
\]

**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:**

\[
\text{LOGSID} = \{ \text{registration_logsid} \\
, \text{listener_node} \\
, \text{log_catalog_name} \\
, \text{log_catalog_instance_name} \\
\}
\]

**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 in a data source when writing to a VSAM or sequential target.

**Operating Systems:** All

**Required:** No

**Syntax:**

\[
\text{LOWVALUES} = \{ \text{N|Y} \}
\]
Valid Values:

- **N**: PowerExchange interprets a hexadecimal '0' value as the end of a string and pads the remaining length of the string with spaces when writing the string to the target.
- **Y**: PowerExchange preserves hexadecimal '0' values when writing a string to the target.

Default is **N**.

Usage Notes:

- To write source hexadecimal '0' values to a z/OS target without translating them, set **LOWVALUES=Y**.
- To ensure that source hexadecimal '0' values are preserved in a PowerCenter session, use the PowerExchange Client for PowerCenter (PWXPC) interface, rather than the PowerExchange ODBC interface.

**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.

**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-00609: Listene103.3.0.111:3720 : Listener 10.33.40.42 -> 10.3.0.1 on port 1234 socket 51PWX-00609 Listene103.3.0.111:3720 : Listener 10.33.40.42 -> 10.3.0.1 on port 1234 socket 51PWX-00609 Listene103.3.0.111:3720 : Listener 10.33.40.42 -> 10.3.0.1 on port 1234 socket 51PWX-00609 Listene103.3.0.111:3720 : Listener 10.33.40.42 -> 10.3.0.1 on port 1234 socket 51

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.

**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 220

### 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]
```
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.
**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 DFSRRC00 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/I 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:

```plaintext
LISTENER=(node1,TCP/IP,9999)
NETPORT=(node1,9999,,"PWX.RUNLIB(IMSJCL)",myps,sub2,sub3,sub4)
```

This sample NETPORT statement passes a PSB called myps 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 171

## 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|ZOSSSL]]
,[,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.
Optional. Size, in bytes, of the data portion of the TCP/IP send buffer. Valid values are from 1024 through 1048576. Default is 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.

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 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.

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.

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.

Optional. PowerExchange uses SSL authentication.

Specify the ZOSSL option when you contact a PowerExchange Listener process on z/OS. Otherwise, specify the SSL option.

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:

\texttt{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 \texttt{NOGETHOSTBYNAME} statement to temporarily change PowerExchange behavior until you resolve the TCP/IP problem.

**NRDB_WRITE_CHAR_NULL_FILL Statement**

The \texttt{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: \texttt{NRDB\_WRITE\_NUM\_NULL\_FILL}

Required: No

Syntax:

\texttt{NRDB\_WRITE\_CHAR\_NULL\_FILL=\{character\|\_\}}

Value: For the \texttt{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 \texttt{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: \texttt{NRDB\_WRITE\_CHAR\_NULL\_FILL}

Required: No

Syntax:

\texttt{NRDB\_WRITE\_NUM\_NULL\_FILL=\{value\|0\}}

Value: For the \texttt{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 LNKLST 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 which Oracle CDC solution is in use for a PowerExchange installation: PowerExchange Express CDC for Oracle or PowerExchange Oracle CDC with LogMiner.

The **ORACLE_CAPTURE_TYPE** value must be consistent with the type of Oracle CAPI_CONNECTION statement that is defined on the system that initiates the connection to the Oracle system for change capture.

To ensure consistent behavior, define the **ORACLE_CAPTURE_TYPE** statement on all systems that are involved in Oracle CDC processing, including the system where the CAPI_CONNECTION statements are defined.

**OPERATING SYSTEMS:** Linux, UNIX, and Windows

**DATA SOURCES:** Oracle

**REQUIRED:** No

**SYNTAX:**

```
ORACLE_CAPTURE_TYPE=[D|L]
```

**VALID VALUES:**

- **D.** PowerExchange uses Express CDC for Oracle and ORAD CAPI_CONNECTION statements.
- **L.** PowerExchange uses Oracle CDC with LogMiner and ORCL CAPI_CONNECTION statements.

No default value is available.

**USAGE NOTES:**

- PowerExchange Express CDC for Oracle handles all character columns as variable length columns, whereas PowerExchange Oracle CDC with LogMiner handles character columns as fixed length or variable length columns. This behavioral difference affects PowerExchange column-level processing and the view of the CDC extraction map that is imported into PowerCenter. As a result, 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 or ORCL CAPI_CONNECTION statements are defined, the CAPI_CONNECTION type implicitly defines the CDC type, and the **ORACLE_CAPTURE_TYPE** statement is optional. However, if other systems are involved in CDC processing, for example, because you run a separate PowerExchange Listener or use offload processing, you must define the **ORACLE_CAPTURE_TYPE** statement in the dbmover.cfg file on each system to explicitly define the CDC type.
You can use only one type of Oracle CAPI_CONNECTION in a dbmover.cfg file. The ORACLE_CAPTURE_TYPE value must be consistent with this CAPI_CONNECTION type. Otherwise, PowerExchange issues an error message and ends abnormally. The following settings are consistent:
- If you use ORAD CAPI_CONNECTION statements, set ORACLE_CAPTURE_TYPE to D.
- If you use ORCL CAPI_CONNECTION statements, set ORACLE_CAPTURE_TYPE to L.

**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 and PowerExchange Oracle CDC with LogMiner 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:**

```plaintext
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:
• “ICUCHCK Utility” on page 286

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 - ORCL and CAPI_CONNECTION - ORAD

Required: Yes, for PowerExchange Oracle CDC with LogMiner and PowerExchange Express CDC for Oracle

Syntax:

```
ORACLEID={collection_id
  ,oracle_db
  [,source_connect_string]
  [,capture_connect_string]
  [,fifth_positional_parameter]
  [,USEDBNAME]
  }
```

Parameters:
**collection_id**

Required. User-defined identifier for this ORACLEID statement. This value must match the ORACOLL parameter value in the ORCL CAPI_CONNECTION statement, the collection ID in the registration group defined for the source tables, and the DBID value in the PowerExchange Logger pwxccl configuration file.

Maximum length is eight characters.

**oracle_db**

Required. Name of the Oracle database that contains the source tables you registered for change data capture.

**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 Oracle CDC with LogMiner and 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, the value of the ORACLE_SID environment variable is used by default. If the ORACLE_SID environment variable is not defined, the default Oracle database is used, if defined.

**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 Oracle CDC with LogMiner or Express CDC with LogMiner. This connection string must be specified in the Oracle Client tnsnames.ora file that is used for connection to the Oracle source database.

If this value is null, the value of the ORACLE_SID environment variable is used by default and the PowerExchange Logger does not use Oracle SQL*Net for connection. If the ORACLE_SID environment variable is not defined, the default Oracle database is used, if defined.

For PowerExchange Oracle CDC with LogMiner or Express CDC for LogMiner, 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 USEDBNAME positional parameter, for example:

```plaintext
ORACLEID=(collection_id,oracle_db,src_connect_string,capture_connect_string,,USEDBNAME)
```

**USEDBNAME**

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:

```sql
select name, db_unique_name from v$database;
```

In this situation, the USEDBNAME 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.

**Operating Systems:** z/OS

**Data Sources:** Adabas

**Related Statements:** SECURITY

**Required:** No

**Syntax:**

```plaintext
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.

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 CA IDMS/DB.

**Operating Systems:** z/OS

**Data Sources:** CA IDMS/DB

**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 CA IDMS/DB 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 CA IDMS/DB.

**Note:** You must define the CA IDMS/DB load libraries in the DTLLOAD DD statement instead of in the STEPLIB DD statement. If you use netport jobs to access CA IDMS/DB data, define the CA IDMS/DB 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 CA IDMS/DB load libraries.

Default is N.

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.
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:

\[ \text{PRGIND} = \{ \text{N} | \text{Y} \} \]

Valid Values:

- \text{N}. PowerExchange does not write read progress messages to the PowerExchange message log file.
- \text{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 session.

Default is \text{N}.

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:

\[ \text{PRGINT} = \{ \text{num\_rows} | \text{250} \} \]

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 \text{Y} for the PRGIND statement.

Valid values are 0 through 2147483647. Default is 250.

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:

\[ \text{PWXSOMAXCONN} = \text{num\_connections} \]

Value: For the \text{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 the FACILITY class, define it in RACF, or in the security product that you use, with a resource name with a maximum length of 128 bytes.

Related Topics:

- "z/OS Security" on page 206

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, 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={F|FB|FBA|FU|V|VB|VBA|VU}
```

Valid Values:

- **F**: 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:

\[ \text{RMTSYSNAME} = \text{host} \_ \text{name} \]

Value: For the \textit{host} \_ \textit{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 CRTPWXENV command during installation, the value in the RMTSYSNAME statement is set to either the value specified in the optional RMTSYSNAME parameter in the CRTPWXENV 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:

\[ \text{SECURITY} = (\{0|1|2\},(\{N|Y\})) \]

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, 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, 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:

{NY}

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.
Usage Notes:

- In the z/OS Installation Assistant, if you click AdvancedParms 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, pwxxcl, 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 MVS log files on z/OS, 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.

SERVICE_TIMEOUT Statement

The SERVICE_TIMEOUT statement 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.

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 waits to receive heartbeat data from the associated Listener Service or Logger Service before shutting down and issuing an error message.

  Enter a number from 1 through 60. Default is 5.

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 RRSAF to connect to DB2. To use RRSAF 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:

\[
\text{SPACE} = \{\text{unit_type|T}, \text{primary_space|0}, \text{secondary_space|0}\}
\]

Parameters:

\text{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:

- \text{C}. Cylinders.
- \text{T}. Tracks.

Default is T.

\{\text{primary\_space|0}\}

Required. Primary space amount in the units that are defined in the \text{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.

\{\text{secondary\_space|0}\}

Required. Secondary space amount in the units that are defined in the \text{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.

For more information about SSL, see Chapter 7, “Secure Sockets Layer Support” on page 231.

Operating Systems: All

Related Statements: SSL_ALLOW_SELF_SIGNED, SSL_CIPHER_LIST, SSL_CONTEXT_METHOD, SSL_REQ_CLNT_CERT, and SSL_REQ_SRVR_CERT

Required: No

Syntax:

\[
\text{SSL} = \{\text{PASS=client, KEY=key, [CALIST=calist|CAPATH=directory}]\}
\]

Parameters: Enter the following keywords and values:

\text{PASS=client}

Required. The pass phrase used to make an SSL connection.
KEY=key

Required. The SSL key used to make an SSL connection.

CALIST=calist

Required if you do not specify the CAPATH option. The Certificate Authority list (CALIST) to make an SSL connection.

CAPATH=directory

Required if you do not specify the CALIST option.

On Linux or UNIX, enter the trusted CA directory of the OpenSSL installation.

On Windows, enter the certs directory of the OpenSSL installation.

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.

Operating Systems: All

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 that the system trusts.
- **Y.** Uses self-signed certificates. Self-signed certificates are not verifiable by following the chain to a trusted certificate authority.

Default is N.

SSL_CIPHER_LIST Statement

The SS_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_SELFSIGNED, 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:

<table>
<thead>
<tr>
<th>OpenSSL Cipher Suite Name</th>
<th>AT-TLS Cipher Suite Name</th>
<th>Hexadecimal Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>DHE-RSA-AES256-SHA</td>
<td>TLS_DHE_RSA_WITH_AES_256_CBC_SHA</td>
<td>39</td>
</tr>
<tr>
<td>DHE-DSS-AES256-SHA</td>
<td>TLS_DHE_DSS_WITH_AES_256_CBC_SHA</td>
<td>38</td>
</tr>
<tr>
<td>AES256-SHA</td>
<td>TLS_RSA_WITH_AES_256_CBC_SHA</td>
<td>35</td>
</tr>
<tr>
<td>EDH-RSA-DES-CBC3-SHA</td>
<td>TLS_DHE_RSA_WITH_3DES_EDE_CBC_SHA</td>
<td>16</td>
</tr>
<tr>
<td>EDH-DSS-DES-CBC3-SHA</td>
<td>TLS_DHE_DSS_WITH_3DES_EDE_CBC_SHA</td>
<td>13</td>
</tr>
<tr>
<td>DES-CBC3-SHA</td>
<td>TLS_RSA_WITH_3DES_EDE_CBC_SHA</td>
<td>0A</td>
</tr>
<tr>
<td>DHE-RSA-AES128-SHA</td>
<td>TLS_DHE_RSA_WITH_AES_128_CBC_SHA</td>
<td>33</td>
</tr>
<tr>
<td>DHE-DSS-AES128-SHA</td>
<td>TLS_DHE_DSS_WITH_AES_128_CBC_SHA</td>
<td>32</td>
</tr>
<tr>
<td>AES128-SHA</td>
<td>TLS_RSA_WITH_AES_128_CBC_SHA</td>
<td>2F</td>
</tr>
</tbody>
</table>

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 PWXUSSL 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 TLSCipherParms configuration statement on the z/OS server machine.
- To avoid the use a 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 SSL or TLS versions that the peer supports for PowerExchange SSL communication.

**Operating Systems:** All

**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:

- **SSLV2**. The peer supports SSL version 2.
- **SSLV23**. The peer supports SSL version 2 or 3.
- **SSLV3**. The peer supports SSL version 3.
- **TLSV1**. The peer supports TLS version 1.

Default is TLSV1.

Usage Notes:

- Select compatible values for the client and server.
- TLSV1, the default, selects the most secure and recent style of handshaking. The other values are provided for exceptional cases.

### 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.

For more information about SSL security, see "SSL Security on a PowerExchange Network" on page 231.

Operating Systems: All

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 Y.

### 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.

For more information about SSL security, see "SSL Security on a PowerExchange Network" on page 231.

Operating Systems: All

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.

**STATS Statement**

The STATS statement controls whether PowerExchange writes SMF statistics records for the PowerExchange Listener to SMF or to a file. This statement also controls whether PowerExchange collects summary 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 statistics are available for PowerExchange Listeners on z/OS and Windows.

**Tip:**

**Operating Systems:** z/OS, Windows

**Required:** No

**Syntax:**

$$\text{STATS}=\left\{ \begin{array}{l}
\text{SMF,record_number[,interval[0]]} \\
\text{FILE,dataset_name[,interval[0]]} \\
\text{MONITOR[,interval[0]]} \\
\text{NONE}
\end{array} \right\}$$

**Parameters:**

**SMF,record_number[,interval[0]]**

PowerExchange writes SMF records to SMF data sets.

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 records to a file.

Enter the following parameters:

*dataset_name*

Required. Name of the sequential data set to which PowerExchange writes the SMF records.

*{interval[0]}*

Optional. Interval, in minutes, after which PowerExchange writes an interval SMF record.

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 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.

Optionally, 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, 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 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. For more information, see the PowerExchange Reference Manual.

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

**Data Sources:** All

**Required:** No

**Syntax:**

```
SUP_FUNC=function_name
```

**Value:** Enter a name of up to eight characters.

Default is PWXFUNC.

**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

**Data Sources:** All

**Required:** No

**Syntax:**

```
SUP_REQUEST=requestor_name
```

**Value:** Enter a name of up to four characters.

Default is PWXR.

**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
Data Sources: All
Required: No
Syntax:
  SUP_SSNAME=subsystem_name
Value: Enter a name of up to eight characters.
Default is PWXLSTNR.

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
Data Sources: All
Required: No
Syntax:
  SUP_SSTYPE=subsystem_type
Value: Enter a name of up to four characters.
Default is PWX.

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 Listener, PowerExchange Logger for Linux, UNIX, and Windows, or PowerExchange Condense process listens for commands.
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, which is one of the following:
  
  - For a PowerExchange Condense process, the service name that you specified in the CONDENSENAME statement in the CAPTPARM configuration member or file.
  - For a PowerExchange Logger for Linux, UNIX, and Windows process, the service name that you specified in the CONDENSENAME statement in the pwxccl.cfg file.
  - For a PowerExchange Listener process, the service name that you specified 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, the service name that you specified in the **Start Parameters** property.
  - If you created the application service through the infacmd pwx CreateListenerService command, the service name that you specified for the -StartParameters option on the command.

- **listen_port**
  
  Required. Unique port number on which the PowerExchange Listener, 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_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.

TEMPLHLQ Statement

The TEMPLHLQ 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 TEMPLHLQ 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:

TEMPLHLQ=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)
TEMHLPQ=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|1}]
    [,FLUSH={flush_interval|20}]
    [,RECLEN={record_length|30}]
    [,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:
PowerExchange uses the PFX value to create the member names of the log files in the PowerExchange data library. The 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:
  \[\text{datalib/Pl} \text{listener\_port(prefixnnn)}\]

- PowerExchange Condense and other PowerExchange jobs use the following file naming convention:
  \[\text{datalib/JO} \text{Bjob\_number(prefixnnn)}\]

These naming conventions include the following variables:

- \(\text{datalib}\) is the PowerExchange data library name specified during PowerExchange installation.
- \(\text{job\_number}\) is the i5/OS job number for the tracing subtask, DTLTRTSK, that runs under PowerExchange Condense or other PowerExchange jobs.
- \(\text{listener\_port}\) is the PowerExchange Listener port number.
- \(\text{nnn}\) is a 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:

\[\text{datalib/P02480(PWXLOG001)}\]
\[\text{datalib/P02480(PWXLOG002)}\]
\[\text{datalib/P02480(PWXLOG003)}\]

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 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:
    \[\text{logpath/prefix/DTLLST1.pl} \text{listener\_port.nnnn.log}\]
  
  - Windows Listener Service:
    \[\text{logpath\prefix\DTLLSTNT.pl} \text{listener\_port.nnnn.log}\]
  
  - Windows Listener:
    \[\text{logpath\prefix\DTLLST1.pl} \text{listener\_port.nnnn.log}\]

- The PowerExchange Logger for Linux, UNIX, and Windows uses the following file-naming convention:
  
  - Linux and UNIX:
    \[\text{logpath/prefix/PWXCL.Tyyyyymmddhhmss.pp} \text{id.nnnn.log}\]
  
  - Windows:
    \[\text{logpath\prefix\PWXCL.Tyyyyymmddhhmss.pp} \text{id.nnnn.log}\]

- For other tasks, PowerExchange uses the following file-naming convention:
Linux and UNIX:

\[
\text{logpath/prefix/module.tyyyymmddhhmss.pid.nnnn.log}
\]

Windows:

\[
\text{logpath\prefix\module.tyyyymmddhhmss.pid.nnnn.log}
\]

The variables represent the following values:

- \text{logpath} is the value of the LOGPATH statement in the dbmover.cfg file.
- \text{listener\_port} is the PowerExchange Listener port number.
- \text{module} is the name of the PowerExchange module that is running, such as DTLURDMO for that utility or DTLODBCDRV for PowerCenter operations.
- \text{nnn} is a sequential number from 001 through 999.
- \text{pid} is the process ID of the PowerExchange task.
- \text{yyyyymmddhhmss} is the timestamp when the file was created.

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:

\[
\text{logpath/PWXLOG/DTLST1.p02480.n001.log}
\]
\[
\text{logpath/PWXLOG/DTLST1.p02480.n002.log}
\]
\[
\text{logpath/PWXLOG/DTLST1.p02480.n003.log}
\]

Maximum length for the PFX value is 210 characters.

Z/OS

PowerExchange uses the PFX value as the high-level qualifier (HLQ) for the sequential alternative log data sets. By default, PowerExchange uses dynamically allocated alternative log data sets. However, you can specify DTLLOGnn DD statements in JCL to allocate the alternative log data sets.

You cannot use DFSMS compression for alternative log data sets.

The log 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:
  \[
  \text{prefix.sysid.Plistener\_port.Nnnn}
  \]
- All other PowerExchange batch jobs and started tasks use the following file naming convention:
  \[
  \text{prefix.job\_name.job\_number.sysid.Nnnn}
  \]

The variables represent the following values:

- \text{job\_name} is the job name of the batch job or started task.
- \text{job\_number} is the JES job number, which begins with JOB for batch jobs and STC for started tasks.
- \text{listener\_port} is the PowerExchange Listener port number.
- \text{nnn} is a sequential number from 001 through 999.
- \text{sysid} is the system ID of the z/OS system on which the batch job or started task runs.

For example, a PowerExchange Listener that runs on system MVS1 with a port number of 2480, a PFX value of PWXLOG, and a FILENUM value of 3 creates the following log files:

\[
\text{PWXLOG.MVS1.P02480.N001}
\]
\[
\text{PWXLOG.MVS1.P02480.N002}
\]
\[
\text{PWXLOG.MVS1.P02480.N003}
\]

Maximum length for the PFX value is 16 characters.
APPEND={N|Y}

Optional. Controls how PowerExchange uses log files when it restarts.

Enter one of the following options:

- **N.** PowerExchange opens a new log file or uses the oldest log file as the log file, completely overwriting it.
  
  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 log files do not exist, PowerExchange uses the first nonexistent 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. 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 the log file.
    
    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, 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 from 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 from 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 from 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 from 1 through 2948. Default is 100.

**Note:** On z/OS, PowerExchange limits the amount of log data it writes to manually allocated log data sets to the SIZE value when the data set exceeds the SIZE value. If the data set is smaller than the SIZE value, the data set size limits the amount of log data PowerExchange writes.

**VIEW=(Y|N)**

Optional. Controls whether PowerExchange periodically closes and reopens the current log file. You can specify this parameter on all operating systems, but it is most useful on z/OS. On z/OS, you cannot see any log records in the log data set until it is closed. On other systems, you can view the log records after PowerExchange flushes them to disk.

Enter one of the following options:

- **N.** PowerExchange does not periodically close and reopen the current log file.
- **Y.** PowerExchange periodically closes and reopens the current log file.

On z/OS, a value of Y causes PowerExchange to periodically close and reopen the log data set based on the FLUSH interval, which enables you to view messages. Also, the following limitations 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:

  ```
  //DTLOG01 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:

   ```
   //DJEJGDG1 JOB (ACCOUNT),'GDG',NOTIFY=SYSIN 
   //IJSTEP01 EXEC PGM=IDCMS 
   //SYSPRIN DD SYSOUT=* 
   //SYSIN DD * 
   DEFINE GDG- 
   (NAME(USER1.V901.TRCGDG)- 
    LIMIT(4)- 
    NOEMPTY- 
    SCRATCH) 
   //STEP2 EXEC PGM=IEFBR14 
   ```
2. In the DBMOVER configuration member, define a TRACING statement that includes VIEW=Y and APPEND=Y. For example:

```
TRACING=(PFX=PFX,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 DTLLOGnn DD statement in the PowerExchange Listener JCL to reference the current generation data set of the GDG, for example:

```
DTLOG01 DD DSN=USER1.V901.TRCGDG{+1},DISP=(NEW,CATLG),
       DCB=(BLKSIZE=32718,LRECL=132,RECFM=VB),
       SPACE=(CYL,(1,1))
```

Use DISP=SHR in this statement.

4. Start the PowerExchange Listener.

PowerExchange begins appending messages at the end of the current generation data set.

RELATED TOPICS:

- "PowerExchange Message Log Options" on page 163
- "Allocation Options for Alternative Log Data Sets on z/OS" on page 165

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_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

**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**.

**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

Data Sources: All

Required: No

Syntax:

```
WORKCLASS=transaction_name
```

Value: Enter a name of up to eight characters.

Default is PWXWORK.

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.

---

**PowerExchange Message Log Options**

PowerExchange uses log files to store informational and error messages about bulk and change data capture (CDC) operations.

You can configure PowerExchange to use the following log files:

- **PowerExchange log file.** The PowerExchange log file is the default log file for all PowerExchange messages. PowerExchange creates this log file, if it does not exist. If the PowerExchange log file becomes full, you must clear it. PowerExchange writes each message, as it occurs, to this single log file.

- **PowerExchange alternative log files.** You can request that PowerExchange use alternative log files by coding the TRACING statement in the DBMOVER configuration file. PowerExchange creates these alternative log files, if they do not exist. PowerExchange rotates through the alternative log files, which eliminates out-of-space conditions. Also, PowerExchange buffers messages and writes them to the disk files based on a customized flush interval, thereby reducing I/O activity for these files.

**PowerExchange Message Log File**

By default, PowerExchange writes all messages to the PowerExchange message log file.

PowerExchange creates this file if it does not exist.

The following table shows the name and default location of the PowerExchange message log file by operating system:

<table>
<thead>
<tr>
<th>System Type</th>
<th>File Name</th>
<th>Location</th>
</tr>
</thead>
<tbody>
<tr>
<td>i5/OS</td>
<td>datalib/LOG(DTLLOG)</td>
<td>A member of the LOG file in the PowerExchange data library, datalib, which PowerExchange creates during installation.</td>
</tr>
<tr>
<td>Linux, UNIX, and Windows</td>
<td>detail.log</td>
<td>A file in the current working directory of the PowerExchange process that is running.</td>
</tr>
<tr>
<td>z/OS</td>
<td>DTLLOG DD statement</td>
<td>A data set that is allocated to the DTLLOG DD statement in the PowerExchange job. This DD statement can point to a permanent data set or to a SYSOUT data set.</td>
</tr>
</tbody>
</table>
Writing messages to a single file means that, 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 alternative log files provide more flexibility and better performance than the original PowerExchange log file.

If you configure PowerExchange to use alternative log files, PowerExchange writes most messages to these log files. PowerExchange writes the initial startup messages to the original PowerExchange log file. After PowerExchange initializes the tracing subtask, it writes messages to the alternative log files only.

PowerExchange alternative log files provide the following benefits over the original PowerExchange log:

- PowerExchange can optionally use multiple files for logging messages. When the current log file is full, PowerExchange switches to the next log file, thereby preventing log-full conditions.
- After opening an alternative log file, PowerExchange buffers messages so PowerExchange can write them to the log file at specified intervals. This buffering reduces the amount of open, close, and write activity on the file to improve performance and resource use.
- You can control the size and number of alternative log files and the frequency at which PowerExchange flushes the log records to the file.

To use alternative log files, specify the TRACING statement in the DBMOVER configuration file.

**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.

PowerExchange uses the following DCB attributes to allocate alternative log data sets on z/OS:

- Block size (BLKSIZE) is 27,998.
- Data set organization (DSORG) is PS.
- Record format (RECFM) is VB.
- Record length (LRECL) is the value of the RECLEN parameter on the TRACING statement.

You can use the following formulas to calculate the primary and secondary space, in blocks, of an alternative log data set:

\[
\text{primary space} = \frac{\text{log \_ size}}{27998} \\
\text{secondary space} = \frac{\text{primary space}}{10}
\]

These formulas include the following variables:

- \( \text{log \_ size} \) is the value that you specify for the SIZE parameter in the TRACING statement, which you convert from MB to bytes.
- 27998 is the block size in bytes.
- \( \text{primary space} \) is the value that you calculate by dividing \( \text{log \_ size} \) by 27,998.

For example, specify 100 for the SIZE parameter, and then complete the following calculations:

\[
\text{primary space} = \frac{104,857,600}{27,998} = 3745 \text{ blocks} \\
\text{secondary space} = \frac{3745}{10} = 374 \text{ blocks}
\]
Allocation Options for Alternative Log Data Sets on z/OS

If you specify the TRACING statement in the DBMOVER configuration file, PowerExchange uses alternative log data sets to log messages from PowerExchange jobs and started tasks. Several options are available for allocating the alternative log data sets.

Use any of the following allocation options:

- **Dynamic allocation.** PowerExchange dynamically allocates log data sets unless you specify a DTLLOGnn statement in the JCL for a PowerExchange job or started task. To name the log data sets, PowerExchange uses the PFX parameter value in the TRACING statement. 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 SYSOUT=* 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 SYSOUT data set. In this case, PowerExchange does not use the PFX, APPEND, FILENUM, and VIEW parameters. However, if you have other jobs and started tasks that use dynamic allocation, PowerExchange honors the PFX, APPEND, FILENUM, and VIEW parameters for those jobs and tasks, if defined.

- **DTLLOGnn DD statements with different data set names:** You can specify DTLLOGnn 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 DTLLOGnn DD statements. PowerExchange uses the same round-robin algorithm for the log data sets as it does with dynamically allocated log data sets. The variable nn is a number from 01 to 99. You can specify up to 99 DTLLOGnn 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) to the JCL.

PowerExchange jobs and started tasks that write to the alternative log files include the PowerExchange Listener, PowerExchange Condense, PowerExchange Agent, and netport jobs. Also, the Adabas, Datacom, IDMS, and IMS log-based ECCRs 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 153

**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,DBCCLASS=sample1,USERID=user,USERID=password,
  CCATALOG=DTLCCATLAOG))
/*
CAPI_CONNECTION= (NAME=DTLUDB2,TYPE=(UDB,CAPINAME=CAPIUDB2))
CAPI_CONNECTION= (NAME=CAPIUDB2,TYPE=(UDB,
  DATABASE=sample2,DBCCLASS=sample2,USERID=user,USERID=password,
  CCATALOG=DTLCCATLAOG))
/*
CAPI_CONNECTION= (NAME=DTLUDB,TYPE=(UDB,CAPINAME=CAPIUDB))
CAPI_CONNECTION= (NAME=CAPIUDB,TYPE=(UDB,
  DATABASE=sample,DBCCLASS=sample,USERID=user,USERID=password,
  CCATALOG=DTLCCATLAOG))
CAPI_SRC_DFLT=(UDB,DTLUDB2)
/*
ORACLED= (NAME=OEMCAP,OEMDB,OEMDB.INFORMatica.COM,OEMDB.INFORMatica.COM)
CAPI_CONNECTION= (NAME=CAPIUDWC,TYPE=(UW,CAPINAME=CAPIORA))
CAPI_CONNECTION= (NAME=CAPIORA,TYPE=(ORCL,ORACOLL=OEMCAP))
```

**CAPI_SRC_DFLT Statements**

You can specify a **CAPI_SRC_DFLT** statement for each source type to identify the default CAPI_CONNECTION statement for that source type.

The **CAPI_SRC_DFLT** statement has the following syntax:

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

The second parameter in the **CAPI_SRC_DFLT** statement must point to a CAPI_CONNECTION statement of a compatible type.
The following table shows, for each CAPI_SRC_DFLT source type, the compatible CAPI_CONNECTION type:

<table>
<thead>
<tr>
<th>CAPI_SRC_DFLT Type</th>
<th>CAPI_CONNECTION Statement Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>AS4</td>
<td>UOWC</td>
</tr>
<tr>
<td>CAPX</td>
<td>CAPX</td>
</tr>
<tr>
<td>DB2, IMS, ADA, IDM, VSAM, VSM, DCM, and IML or IDL</td>
<td>UOWC</td>
</tr>
<tr>
<td>ORA</td>
<td>UOWC</td>
</tr>
<tr>
<td>MSS</td>
<td>MSQL</td>
</tr>
<tr>
<td>UDB</td>
<td>UDB</td>
</tr>
</tbody>
</table>

For example, the following statements identify the default CAPI_CONNECTION statements for DB2 for Linux, UNIX, and Windows and for Oracle:

```plaintext
CAPI_SRC_DFLT=(UDB, DLT, UDB2)
CAPI_SRC_DFLT=(ORA, CAPIORA)
```

**Related Topics:**
- "STATS Statement" on page 144

**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:

<table>
<thead>
<tr>
<th>Source System</th>
<th>CAPI_CONNECTION Types</th>
</tr>
</thead>
</table>
| All z/OS sources             | - LRAP CAPI_CONNECTION for the Log Read API that extracts change data from PowerExchange Logger for MVS log files (Required)  
                              | - UOWC CAPI_CONNECTION for the UOW Cleanser (Required)                                    |
| DB2 for i5/OS                | - 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 | - UDB CAPI_CONNECTION for change extraction from DB2 recovery logs (Required)  
                                      | - CAPX CAPI_CONNECTION, if you use continuous extraction mode (Optional)               |
| Microsoft SQL Server         | - MSQL CAPI_CONNECTION for change extraction from Microsoft SQL Server distribution databases (Required)  
                              | - CAPX CAPI_CONNECTION, if you use continuous extraction mode (Optional)               |
| Oracle                       | - ORCL CAPI_CONNECTION for change extraction from Oracle redo logs (Required)  
                              | - UOWC CAPI_CONNECTION for the UOW Cleanser (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:

<table>
<thead>
<tr>
<th>Statement</th>
<th>Description</th>
<th>Reference</th>
</tr>
</thead>
<tbody>
<tr>
<td>LISTENER</td>
<td>Required. Defines the TCP/IP port on which a named PowerExchange Listener process listens for work requests.</td>
<td>&quot;LISTENER Statement&quot; on page 109</td>
</tr>
<tr>
<td>SECURITY</td>
<td>Optional. Controls whether PowerExchange authenticates user access to PowerExchange and authorizes users to issue specific pwxcmd commands.</td>
<td>&quot;SECURITY Statement&quot; on page 195</td>
</tr>
<tr>
<td>SVCNODE</td>
<td>Required. Specifies the TCP/IP port on which a PowerExchange Listener process listens for commands.</td>
<td>&quot;SVCCNODE Statement&quot; on page 147</td>
</tr>
</tbody>
</table>

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:

<table>
<thead>
<tr>
<th>Statement</th>
<th>Configuration File</th>
<th>Description</th>
<th>Reference</th>
</tr>
</thead>
<tbody>
<tr>
<td>CONDENSENAME</td>
<td>pwxccl.cfg file</td>
<td>Required. Defines a name for the command-handling service for a PowerExchange Logger process that is the target of pwxcmd commands.</td>
<td>CDC guide for the operating system</td>
</tr>
<tr>
<td>SECURITY</td>
<td>DBMOVER configuration file</td>
<td>Optional. Controls whether PowerExchange authenticates user access to PowerExchange and authorizes users to issue specific pwxcmd commands.</td>
<td>“SECURITY Statement” on page 195</td>
</tr>
<tr>
<td>SVCNODE</td>
<td>DBMOVER configuration file</td>
<td>Required. Specifies the TCP/IP port on which a PowerExchange Logger for Linux, UNIX, and Windows process listens for commands.</td>
<td>“SVCNODE Statement” on page 147</td>
</tr>
</tbody>
</table>

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:

<table>
<thead>
<tr>
<th>Statement</th>
<th>Configuration File</th>
<th>Description</th>
<th>Reference</th>
</tr>
</thead>
<tbody>
<tr>
<td>CMDNODE</td>
<td>dbmover.cfg file</td>
<td>Required. Specifies connection information for a PowerExchange process that is the target of pwxcmd commands.</td>
<td>“CMDNODE Statement” on page 75</td>
</tr>
</tbody>
</table>

Sample DBMOVER Configuration Files

PowerExchange provides an example DBMOVER configuration file for each system.

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 datalib library
- On Linux, UNIX, and Windows, the 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.
Netport Jobs

This chapter includes the following topics:

• Netport Jobs Overview, 171
• Configuring Netport Jobs, 172
• Sample Netport Jobs, 172
• PowerExchange Substitution Variables for Netport Jobs, 173
• Netport Jobs and Offload Processing, 177
• Netport Jobs and DB2 Bulk Data Loads, 177
• Using Netport Jobs for Generation Data Sets, 177

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 generation data sets (GDGs). In this case, special considerations and configuration requirements apply. See “Using Netport Jobs for Generation Data Sets” on page 177.

• Access IDMS data when full user ID checking is required, such as when the SECURITY statement in the DBMOVER configuration file is set to 2,x.

To link a netport job to a port, you must define a pair of NETPORT and LISTENER statements in the DBMOVER configuration file.
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,,"UTILSR.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 173.

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:

```jcl
//SYSIDMS DD DSN=6HLQ..RUNLIB(DTLDLCL), DISP=(SHR)
```

Ensure that the relevant SYSCTL statement is included in the JCL for Central Versions, or the following DD statements if running local:

```jcl
//IDMSDICT INCLUDE MEMBER=IDMSDICT
//IDMSFIL INCLUDE MEMBER=IDMSFIL
```

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 172

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:

<table>
<thead>
<tr>
<th>Substitution Variable</th>
<th>Where Used in the JCL</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>%CLIENT</td>
<td>Fourth positional parameter on the first line of the PARMS statement</td>
<td>Unique 16-byte hexadecimal identifier that includes the job name of the PowerExchange Listener.</td>
</tr>
<tr>
<td>%CPNODE</td>
<td>First positional parameter on the first line of the PARMS statement</td>
<td>The cpnode parameter value from the NETPORT statement that PowerExchange used to submit the netport job.</td>
</tr>
<tr>
<td>%CPPORT</td>
<td>Third positional parameter on the first line of the PARMS statement</td>
<td>The cpport parameter value from the NETPORT statement that PowerExchange used to submit the netport job.</td>
</tr>
<tr>
<td>%DATE_YYMMDD</td>
<td>DD statement</td>
<td>Current Gregorian date.</td>
</tr>
<tr>
<td>%DATE_YYYYDDD</td>
<td>DD statement</td>
<td>Current Julian date.</td>
</tr>
</tbody>
</table>
| %DMX_ECSA             | Keyword parameter on the second line of the PARMS statement | The hexadecimal address of the ECSA memory that holds the time of the latest update to the DATAMAPS file. Use the DMX_ECSA keyword parameter to enter this substitution variable on the second line in the PARMS DD, as follows: 

```
DMX_ECSA=%DMX_ECSA
```

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.

If you do not use data maps caching, this parameter is ignored. |
<p>| %IMSID                | - | A substitution variable for the IMS SSID. if you specify an IMS SSID override in the PowerCenter <strong>IMS SSID Override</strong> session property for a source or target or in the <strong>IMS SSID</strong> advanced parameter for a database row test, the override value replaces the substitution variable for the session or row test. 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. 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>
<table>
<thead>
<tr>
<th>Substitution Variable</th>
<th>Where Used in the JCL</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>%Nn</td>
<td>JOB statement</td>
<td>Value of the numeric counter with the length of n. Valid values for the length are from 1 through 7. 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.</td>
</tr>
<tr>
<td>%PSBNAME</td>
<td>-</td>
<td>A substitution variable for the PSB name. 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. 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. <strong>Note:</strong> 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. 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: <code>//DBALLOC INCLUDE MEMBER=%PSBNAME</code> By using this variable with PSB overrides, you can allocate other database members to the job.</td>
</tr>
<tr>
<td>%PWD</td>
<td>JOB statement</td>
<td>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.</td>
</tr>
<tr>
<td>%SOCKNO</td>
<td>Second positional parameter on the first line of the PARMS statement</td>
<td>Socket number on which the connection to the PowerExchange Listener was made.</td>
</tr>
<tr>
<td>%SMFTASK</td>
<td>-</td>
<td>If the STATS statement in the DBMOVER member specifies recording to SMF, this variable is the PowerExchange task number. Otherwise, the value is 00000.</td>
</tr>
<tr>
<td>%TIME_HHMMSS</td>
<td>DD statement</td>
<td>Current time, in 24-hour clock time format.</td>
</tr>
<tr>
<td>%USER</td>
<td>JOB statement</td>
<td>The user ID that connected to the PowerExchange Listener.</td>
</tr>
<tr>
<td>Substitution Variable</td>
<td>Where Used in the JCL</td>
<td>Description</td>
</tr>
<tr>
<td>-----------------------</td>
<td>-----------------------</td>
<td>-------------</td>
</tr>
<tr>
<td>%1</td>
<td>-</td>
<td>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.</td>
</tr>
<tr>
<td>%2</td>
<td>-</td>
<td>An optional substitution variable. Enter it as the seventh positional parameter in the NETPORT statement in the DBMOVER member.</td>
</tr>
<tr>
<td>%3</td>
<td>-</td>
<td>An optional substitution variable. Enter it as the eighth positional parameter in the NETPORT statement in the DBMOVER member.</td>
</tr>
<tr>
<td>%4</td>
<td>-</td>
<td>An optional substitution variable. Enter it as the ninth positional parameter in the NETPORT statement in the DBMOVER member.</td>
</tr>
<tr>
<td>%5</td>
<td>-</td>
<td>An optional substitution variable. Enter it as the tenth positional parameter in the NETPORT statement in the DBMOVER member.</td>
</tr>
<tr>
<td>%6</td>
<td>-</td>
<td>An optional substitution variable. Enter it as the eleventh positional parameter in the NETPORT statement in the DBMOVER member.</td>
</tr>
<tr>
<td>%7</td>
<td>-</td>
<td>An optional substitution variable. Enter it as the twelfth positional parameter in the NETPORT statement in the DBMOVER member.</td>
</tr>
<tr>
<td>%8</td>
<td>-</td>
<td>An optional substitution variable. Enter it as the thirteenth positional parameter in the NETPORT statement in the DBMOVER member.</td>
</tr>
</tbody>
</table>

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=DTLLST31,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 PSBNOME 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 the 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:

```java
//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)
//SYSSERR 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. Using a relative generation number, you can reference the most current generation data set in the GDG using a fixed data set name.

This practice is extremely useful for PowerExchange data maps and partner ETL tools. The GDG name can be put in a single data map without requiring dynamic updates to the data map data set name. You can also write to GDG data sets using the relative generation number.

When you run a batch job and access a GDG data set using a relative generation number, such as AAA.BBB.CCC(0), you access the same generation data set for the life of the batch job regardless of whether more current generations exist. This is a good way to handle things for z/OS batch jobs. With long running
batch jobs or with started tasks such as the PowerExchange Listener, you normally want to access the current generation data set when using relative references.

By default, the PowerExchange Listener does not refresh the generation table after the first reference to a generation data set. After the PowerExchange Listener accesses a generation data set using a relative generation number, all future references to that GDG access the same generations as the first reference. As a result, you cannot reference any newly created generations.

To have the PowerExchange Listener refresh the generation table when using relative generation references, choose one of the following options:

- Enter GDGLOCATE=Y in the DBMOVER configuration member. This setting will ensure that the catalog is referenced when access to a GDG is requested (reading an existing data set or creating a new data set) to make the latest information available. This feature prevents the need for recycling the PowerExchange Listener.

- Set up a NETPORT job for GDG types to allow the latest GDGs to be referenced. This practice is similar to how IMS is handled. Each time the file is accessed, the NETPORT job shuts down and a subsequent call is made to invoke another NETPORT job. All GDGs are recognized as you intended. NETPORT jobs are used for IMS, IDMS, and for accessing tapes with long mount times. Ensure that the NETPORT job runs on the same z/OS image as the PowerExchange Listener to satisfy socket API call constraints. Otherwise, the jobs might timeout.

- Recycle the PowerExchange Listener at regular intervals, more frequently than the GDGs are created or need to be created by means of PowerExchange.

Setting Up Netport Jobs for GDGs

Use this procedure to set up a netport job for GDGs.

To set up netport jobs for GDGs:

1. Determine an available z/OS port number (“7777” in this example) to use with the netport job.
2. Add a NODE statement to your source and target DBMOVER configuration files, for example:

   ```
   NODE=(MVS,TCPIP,mvs1,2480)
   NODE=(MVSDDG,TCPIP,mvs1,7777)
   ```

   If you are using 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:

   ```
   LISTENER=(node1,TCPIP,2480)
   LISTENER=(node1,TCPIP,7777)
   ```

   To indicate to the z/OS PowerExchange Listener that the second port is not typical, add a NETPORT statement with the appropriate node name and port number. For example:

   ```
   NETPORT=(node1,7777,,"INFA.Vxyz.RUNLIB(GDGJCL)",,)
   ```

4. Create a member in the RUNLIB library called GDGJCL, using the TAPEJCL member as a template.

   When the z/OS PowerExchange Listener detects activity on the netport, instead of directly trying to service it, it tries to submit the JCL file named in the fifth token of the NETPORT statement to the JES internal reader. As it is reading and submitting the JCL, it scans every JCL line for tokens %x, where x is a numeric value from 1 through 8. It substitutes these tokens with the appropriate value taken from parameters 6 through 13 of the NETPORT statement. Do not confuse the %x tokens with the %Nx tokens. The %Nx tokens are used to generate incremented numeric values of x length.

   To create the GDGJCL job from the TAPEJCL job, add a job card leaving "%N5" appended to it and any other DDs that you might need for your PowerExchange Listener job. You can then submit the GDGJCL
job whenever a request is received through the assigned port for the netport job. This enables a job to end and another job to begin and handle 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 the z/OS location.
CHAPTER 4

PowerExchange Message Destination Override

This chapter includes the following topics:

- PowerExchange Message Destination Override Overview, 180
- DTLMSGO File, 181

PowerExchange Message Destination Override Overview

The text for the majority of PowerExchange messages is contained within the DTLMSG file, which is created during the installation process using the contents shipped with the product. PowerExchange displays these messages in several possible locations:

- PowerExchange log file (DTLLOG or an alternative log file)
- Standard output (stdout)
- System operator console (z/OS and i5/OS only)

Using the message destination override capability, you can determine to which, if any, of these locations a particular message is routed. This capability is useful for redirecting messages to destinations other than their default ones when you want to enable automation or adhere to installation-specific output handling standards. This capability also enables you to suppress specific messages. Before suppressing messages, consider the implications. Usually, destination overrides are used for only a small number of messages.

To configure Message Destination Override, create a file called DTLMSGO. In this file, specify the message number followed by the destinations to which the message is to be written. The manner in which the location of the DTLMSGO file is specified to PowerExchange varies by system.

Note: On z/OS, messages other than those in the DTLMSG file can be issued. These messages are composed of the messages written 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, Datacom, DB2 for z/OS, IDMS synchronous, and IMS synchronous ECCRs
DTLMSGO File

The location of the DTLMSGO file varies by system:

- On z/OS, the JCL must contain a DTLMSGO DD statement that points to a sequential data set or PDS member. A sample file is provided in the DTLMSGO member of the RUNLIB library.
- 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 installation directory. A sample file, called dtlmsgo_sample.txt, is provided in the installation directory.

Syntax Rules for DTLMSGO Statements

Use the following syntax rules when defining DTLMSGO statements:

- Comment lines must begin with "/*" in column 1.
- If the same message appears more than once in the DTLMSGO file, the last occurrence is used.

DTLMSGO Statement Syntax

Use the following syntax for DTLMSGO statements:

```
msg_number, {msg_dest}, {msg_dest}, ... | msg_number
```

DTLMSGO Statement Required Parameter

The following parameter is required in DTLMSGO statements:

`msg_number`

This is the message number from the DTLMSG file (dtlmsg.txt on UNIX/Windows). The message numbers are 5-digit numbers with no prefix.

No generic or masking capability exists so complete message numbers must be specified, although it is not necessary to specify the leading zeroes.

DTLMSGO Statement Optional Parameters

The following parameters are optional in DTLMSGO statements:

`msg_dest`

Valid values for this parameter are:

- `null`. If no destination is specified, the message is not rerouted to a different destination. Rather, the message is treated as if it is not included in the DTLMSGO file.
- `LOG`. Specifies that the message is directed to the PowerExchange Log. This is either the DTLLOG file or the alternative logging file. If alternative logging is used, the message might be redirected to the DTLLOG file if it is issued prior to the alternative logging subtask being initialized.
- **NONE.** Specifies that the message is suppressed. The message is not displayed in any of the other locations. This parameter is mutually exclusive with other `msg_dest` values.

- **STDOUT.** Specifies that the message is directed to the standard output location, which varies by system.
  
  On z/OS, STDOUT is sent to the SYSPRINT DD if specified or a dynamically allocated temporary JES SPOOL data set if not specified.
  
  On i5/OS, STDOUT is sent to QPRINT if PowerExchange is running as a background job or to the terminal if it is running interactively.
  
  On UNIX and Windows, STDOUT is sent to the terminal by default unless you have redirected it elsewhere.

- **SYSCONS.** On z/OS, specifies that the message is displayed by the Write-To-Operator (WTO) macro. No routing code is associated with the WTO so the ROUTCODE value specified in the DEFAULT statement of CONSOLxx is used for routing.
  
  On i5/OS, SYSCONS results in the message being sent to the operator message queue.
  
  On Linux, UNIX, and Windows, SYSCONS is ignored.

**DTLMSGO File Example**

The following example DTLMSGO file shows how to code the following types of messages: messages without leading zeroes (594 and 607), messages without any change in destination (00595), and messages with multiple destinations (00650, 00651, and 607):

```bash
/*
 /* Sample file to change message destinations
 /*
 /* region size mag to console
 594,SYSCONS
 /* use pwx defaults
 00595
 /* multiple destinations
 00650,SYSCONS,LOG
 00651,SYSCONS,LOG,STDOUT
 /* suppress stats mag
 00408,NONE
 /*
 607,STDOUT,LOG
 /*
```
Statistics Logging Using SMF

This chapter includes the following topics:

- Statistics Logging Using SMF Overview, 183
- Configuring SMF Logging on z/OS, 184
- Configuring SMF Logging to a File, 184
- SMF Record Format, 186

Statistics Logging Using SMF Overview

PowerExchange provides statistics records on z/OS that you can log by using IBM System Management Facilities (SMF).

PowerExchange provides the following types of statistics records:

- Connection record driven by the PowerExchange Listener
- Start record driven by the PowerExchange process, netport, PowerExchange Listener, and subtask
- Interval statistics record for all processes
- End record for all processes, including processes that abnormally end

The following information is collected:

- General connection information
- General resource measurements
- Access method specific information
- Access method specific measurements

Configure the record number for which to log statistics data in the STATS statement in the DBMOVER configuration file. Consult with your system programmer or capacity planning team to choose this record number.

You can write statistics data to SMF or, for testing, to a flat file.

PowerExchange provides interval logging through the interval statistics record. This record shows the cumulative statistics for all tasks that are active at a specified time. Use these statistics to monitor the activity and progress of long running tasks. By examining the statistics in two interval reports for a specific task, you can determine the activities for the task for a specified time interval.
Interval logging reports the statistics totals for tasks that are active when PowerExchange reports the interval statistics. Interval logging does not report on the following types of activities:

- Activity by tasks that started and completed between the two interval reports
- Activity by a task that ended since the last interval report

### Configuring SMF Logging on z/OS

To perform SMF logging on z/OS, specify the STATS statement with the SMF parameter in the DBMOVER configuration file on z/OS. Then, restart the PowerExchange Listener. By default, PowerExchange does not collect statistics.

The STATS statement with the SMF parameter uses the following syntax:

```plaintext
STATS=(SMF,record_number,{interval|0})
```

The following table describes the parameters:

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>record_number</code></td>
<td>SMF record number. Valid values are from 128 through 255.</td>
</tr>
<tr>
<td><code>interval</code></td>
<td>Interval in minutes. Valid values are 0 and any number from 5 through 120. Specify 0 to disable interval processing. PowerExchange writes an SMF record when the PowerExchange Listener shuts down. Default is 0.</td>
</tr>
</tbody>
</table>

For example:

```plaintext
STATS=(SMF,255,0)
```

**Note:** To enable PowerExchange to write to SMF, you must APF-authorize all libraries in the STEPLIB of the PowerExchange Listener.

### Configuring SMF Logging to a File

To get statistics quickly during testing, you can write statistics records to a file instead of to SMF.

To log statistics to a file, specify the STATS statement with the FILE parameter in the DBMOVER configuration file. Then, restart the PowerExchange Listener. By default, PowerExchange does not collect statistics.

The STATS statement with the FILE parameter uses the following syntax:

```plaintext
STATS=(FILE,filename,{interval|0})
```
The following table describes the parameters:

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>filename</td>
<td>Name of the file to which to write statistics.</td>
</tr>
<tr>
<td>interval</td>
<td>Interval in minutes. Valid values are 0 and any number from 5 through 120. Specify 0 to disable interval processing. PowerExchange writes an SMF record when the PowerExchange Listener shuts down. Default is 0.</td>
</tr>
</tbody>
</table>

For example:

```
STATS=(FILE,DTLSR.V800.SMFFILE,0)
```

**Viewing Statistics Written to a File**

You can use the sample pwxstat.file data map to view statistics that you write to a file.

The pwxstat.file data map file is in the examples\datamaps subdirectory in the PowerExchange installation directory on the PowerExchange Navigator system.

To view statistics written to a file:

1. On z/OS, stop the PowerExchange Listener, and edit the DBMOVER configuration file to include a STATS statement with the FILE parameter. In the STATS statement, specify the name of a z/OS data set to which you want to write statistics.
   
   For example:
   
   ```
   STATS=(FILE,DTLSR.V800.SMFFILE,0)
   ```
   
   Then, start the PowerExchange Listener.

2. In the PowerExchange Navigator, open the pwxstat.file data map.


4. Click the SEQ Access Method tab in the Data Map Properties dialog box.

5. In the File Name box, enter the name of the z/OS data set that you specified in the STATS statement in the DBMOVER configuration file on z/OS.

6. Click OK.

7. To generate statistics, on z/OS, stop and start the PowerExchange Listener.

8. In the PowerExchange Navigator, run a database row test on any of the tables in the pwxstat.file data map.

9. In the Database Row Test dialog box, select NRDB in the DB Type box. Select the node location of the z/OS system in the Location list.

10. Click Go.

   **Warning:** If you write statistics directly to a single sequential file or GDG, operational problems can occur in the PowerExchange environment.
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:

<table>
<thead>
<tr>
<th>Sub Type</th>
<th>General Section</th>
<th>Extended Section</th>
<th>Function/Access Method Section</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>X</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>2</td>
<td>X</td>
<td>-</td>
<td>X</td>
</tr>
<tr>
<td>3</td>
<td>X</td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td>4</td>
<td>X</td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td>13</td>
<td>X</td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td>14</td>
<td>X</td>
<td>X</td>
<td>X</td>
</tr>
</tbody>
</table>

Standard SMF Header with Subtype

The following table describes the fields in the standard SMF header with subtype section:

<table>
<thead>
<tr>
<th>Field</th>
<th>Decimal Offset</th>
<th>Hexadecimal Offset</th>
<th>Length</th>
<th>Format</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>SMFxLEN</td>
<td>0</td>
<td>00</td>
<td>2</td>
<td>Binary</td>
<td>Record length. The SMFxLEN and SMFxSEG fields, which total four bytes, form the record descriptor word (RDW). The first two bytes (SMFxLEN) contain the logical record length.</td>
</tr>
<tr>
<td>SMFxSEG</td>
<td>2</td>
<td>02</td>
<td>2</td>
<td>Binary</td>
<td>Segment descriptor, which is used for variable block spanned records. If the record is not spanned, these two bytes are set to hexadecimal zeros.</td>
</tr>
<tr>
<td>SMFxFLG</td>
<td>4</td>
<td>04</td>
<td>1</td>
<td>Binary</td>
<td>Header flag byte.</td>
</tr>
<tr>
<td>SMFxRTY</td>
<td>5</td>
<td>05</td>
<td>1</td>
<td>Binary</td>
<td>Record type. Hexadecimal values are from 0 through FF.</td>
</tr>
</tbody>
</table>
### SMFxTME
- **Decimal Offset**: 6
- **Hexadecimal Offset**: 06
- **Length**: 4
- **Format**: Binary
- **Description**: 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.

### SMFxDTE
- **Decimal Offset**: 10
- **Hexadecimal Offset**: 0A
- **Length**: 4
- **Format**: Packed
- **Description**: Date when the record was moved into the SMF buffer, in the form 00 yyddd F or 0 cyyddd F. Where:
  - c is 0 for 19xx, and 1 for 20xx.
  - yy is the current year from 0 through 99.
  - dddd 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
- **Decimal Offset**: 14
- **Hexadecimal Offset**: 0E
- **Length**: 4
- **Format**: EBCDIC
- **Value**: System ID, taken from the SID parameter.

### SMFxSSI
- **Decimal Offset**: 18
- **Hexadecimal Offset**: 12
- **Length**: 4
- **Format**: EBCDIC
- **Value**: Subsystem identification. This field is a four-byte character value set by the SUBSYS=option specified in the SMF macros.

### SMFxSTY
- **Decimal Offset**: 22
- **Hexadecimal Offset**: 16
- **Length**: 2
- **Format**: Binary
- **Value**: Record subtype. Hexadecimal values are from 0 through FF.

For more information about the standard SMF header with record subtypes, see the IBM guide, z/OS V1R7.0 MVS System Management Facilities (SMF).

### PowerExchange Header/Descriptor
The following table describes the fields in the PowerExchange header/descriptor section:

<table>
<thead>
<tr>
<th>Field</th>
<th>Decimal Offset</th>
<th>Hexadecimal Offset</th>
<th>Length</th>
<th>Format</th>
<th>Value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Eye catcher</td>
<td>0</td>
<td>0</td>
<td>8</td>
<td>Char</td>
<td>PWXHDR_</td>
<td>Eye catcher.</td>
</tr>
<tr>
<td>Starting clock time</td>
<td>8</td>
<td>8</td>
<td>8</td>
<td>Binary</td>
<td>-</td>
<td>Starting clock time.</td>
</tr>
<tr>
<td>Ending clock time</td>
<td>16</td>
<td>10</td>
<td>8</td>
<td>Binary</td>
<td>-</td>
<td>Ending clock time.</td>
</tr>
</tbody>
</table>
### Extended Section Descriptor

The following table describes the fields in the extended section descriptor:

<table>
<thead>
<tr>
<th>Field</th>
<th>Decimal Offset</th>
<th>Hexadecimal Offset</th>
<th>Length</th>
<th>Format</th>
<th>Value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Section type</td>
<td>0</td>
<td>0</td>
<td>4</td>
<td>Binary</td>
<td></td>
<td>Numeric value that represents the section type.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>1</td>
<td>PowerExchange Listener section</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>2</td>
<td>Exception section</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>3</td>
<td>File section</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>4</td>
<td>DB2 section</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>5</td>
<td>Client section</td>
</tr>
<tr>
<td>Section count</td>
<td>4</td>
<td>4</td>
<td>4</td>
<td>Binary</td>
<td></td>
<td>Count of sections that follow of the same type.</td>
</tr>
<tr>
<td>Section length</td>
<td>8</td>
<td>8</td>
<td>4</td>
<td>Binary</td>
<td></td>
<td>Length of sections that follow.</td>
</tr>
</tbody>
</table>

### General Section

The following table describes the fields in the general section:

<table>
<thead>
<tr>
<th>Field</th>
<th>Decimal Offset</th>
<th>Hexadecimal Offset</th>
<th>Length</th>
<th>Format</th>
<th>Value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Sub type</td>
<td>0</td>
<td>0</td>
<td>2</td>
<td>Binary</td>
<td></td>
<td>Numeric value that represents the subtype.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>x'0001' = Connection request</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>x'0002' = Start</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>x'0003' = Interval</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>x'0004' = End</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>x'0013' = PowerExchange Listener interval</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>x'0014' = PowerExchange Listener end (same as in the standard SMF header)</td>
<td></td>
</tr>
<tr>
<td>Product</td>
<td>2</td>
<td>2</td>
<td>14</td>
<td>Char</td>
<td></td>
<td>PowerExchange product.</td>
</tr>
<tr>
<td>Version</td>
<td>16</td>
<td>10</td>
<td>10</td>
<td>Char</td>
<td></td>
<td>The PowerExchange version and build.</td>
</tr>
<tr>
<td>Field</td>
<td>Decimal Offset</td>
<td>Hexadecimal Offset</td>
<td>Length</td>
<td>Format</td>
<td>Value</td>
<td>Description</td>
</tr>
<tr>
<td>---------------------</td>
<td>----------------</td>
<td>--------------------</td>
<td>--------</td>
<td>--------</td>
<td>-------</td>
<td>-------------</td>
</tr>
<tr>
<td>Component</td>
<td>26</td>
<td>1A</td>
<td>16</td>
<td>Char</td>
<td>String, such as Listener</td>
<td>The name of the component, such as Listener, the utility name, and so on.</td>
</tr>
<tr>
<td>Job Name</td>
<td>42</td>
<td>2A</td>
<td>8</td>
<td>Char</td>
<td>-</td>
<td>PowerExchange Listener or the name of the process.</td>
</tr>
<tr>
<td>Node name</td>
<td>50</td>
<td>32</td>
<td>8</td>
<td>Char</td>
<td>-</td>
<td>Node name, when applicable.</td>
</tr>
<tr>
<td>Task or process ID</td>
<td>58</td>
<td>3A</td>
<td>5</td>
<td>Char</td>
<td>A value from 1 through 99999</td>
<td>The task or process ID.</td>
</tr>
<tr>
<td>Reserved</td>
<td>63</td>
<td>3F</td>
<td>3</td>
<td>Binary</td>
<td>-</td>
<td>Reserved for future use.</td>
</tr>
<tr>
<td>Client Session ID</td>
<td>74</td>
<td>42</td>
<td>8</td>
<td>Char</td>
<td>Value from the SESSID parameter of the connection parameters. For ODBC, the DTLSESSID SQL escape sequence.</td>
<td>Session ID of the client.</td>
</tr>
<tr>
<td>Client IP address</td>
<td>78</td>
<td>4E</td>
<td>128</td>
<td>Char</td>
<td>Value from dbcb</td>
<td>IP address of the client.</td>
</tr>
<tr>
<td>Client user ID</td>
<td>206</td>
<td>CE</td>
<td>20</td>
<td>Timestamp</td>
<td>-</td>
<td>The user ID for the client.</td>
</tr>
<tr>
<td>Start Time</td>
<td>226</td>
<td>E2</td>
<td>2</td>
<td>Binary</td>
<td>-</td>
<td>Alignment field.</td>
</tr>
<tr>
<td>End Time2</td>
<td>228</td>
<td>E4</td>
<td>20</td>
<td>Timestamp</td>
<td>-</td>
<td>The ending time. For DB2, the format is: YYYYMMDDHHMMSSNNNNNNN</td>
</tr>
<tr>
<td>Reserved</td>
<td>248</td>
<td>F8</td>
<td>2</td>
<td>Binary</td>
<td>-</td>
<td>Alignment field.</td>
</tr>
<tr>
<td>Return Code</td>
<td>250</td>
<td>FA</td>
<td>8</td>
<td>Char</td>
<td>-</td>
<td>Return code.</td>
</tr>
<tr>
<td>Reason Code</td>
<td>258</td>
<td>102</td>
<td>8</td>
<td>Char</td>
<td>-</td>
<td>Reason code.</td>
</tr>
</tbody>
</table>
### Extended Section for the PowerExchange Listener

The following table describes the fields in the extended section for the PowerExchange Listener:

<table>
<thead>
<tr>
<th>Field</th>
<th>Decimal Offset</th>
<th>Hexadecimal Offset</th>
<th>Length</th>
<th>Format</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Additional Info</td>
<td>266</td>
<td>10A</td>
<td>24</td>
<td>Char</td>
<td>-</td>
</tr>
<tr>
<td>Reserved</td>
<td>290</td>
<td>122</td>
<td>32</td>
<td>Binary</td>
<td>-</td>
</tr>
<tr>
<td>CPU Time</td>
<td>318</td>
<td>13E</td>
<td>8</td>
<td>Binary</td>
<td>-</td>
</tr>
<tr>
<td>Descriptors length</td>
<td>326</td>
<td>146</td>
<td>4</td>
<td>Binary</td>
<td>-</td>
</tr>
</tbody>
</table>

For a nonzero return code, additional information about the error.

Reserved for future use.

CPU time, in microseconds.

Length of descriptor sections.

<table>
<thead>
<tr>
<th>Field</th>
<th>Decimal Offset</th>
<th>Hexadecimal Offset</th>
<th>Length</th>
<th>Format</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Job/STC name</td>
<td>0</td>
<td>0</td>
<td>8</td>
<td>Char</td>
<td>Name of job or started task.</td>
</tr>
<tr>
<td>Nodename</td>
<td>8</td>
<td>8</td>
<td>8</td>
<td>Char</td>
<td>Node name.</td>
</tr>
<tr>
<td>Ip Address</td>
<td>16</td>
<td>10</td>
<td>16</td>
<td>Char</td>
<td>IP address.</td>
</tr>
<tr>
<td>Start time</td>
<td>32</td>
<td>20</td>
<td>20</td>
<td>Timestamp</td>
<td>The time stamp. For DB2, the format is: YYYMMDDHHMMSSNNNNNN</td>
</tr>
<tr>
<td>End time</td>
<td>52</td>
<td>34</td>
<td>20</td>
<td>Timestamp</td>
<td>The ending time. For DB2, the format is: YYYMMDDHHMMSSNNNNNN</td>
</tr>
<tr>
<td>STCK start</td>
<td>72</td>
<td>48</td>
<td>8</td>
<td>Binary</td>
<td>Starting time (STCK).</td>
</tr>
<tr>
<td>STCK end</td>
<td>80</td>
<td>50</td>
<td>8</td>
<td>Binary</td>
<td>Ending time (STCK).</td>
</tr>
<tr>
<td>Port Number</td>
<td>88</td>
<td>58</td>
<td>4</td>
<td>Binary</td>
<td>Port number.</td>
</tr>
<tr>
<td>Number connections</td>
<td>92</td>
<td>5C</td>
<td>4</td>
<td>Binary</td>
<td>The number of connections.</td>
</tr>
<tr>
<td>Reserved</td>
<td>96</td>
<td>60</td>
<td>4</td>
<td>Binary</td>
<td>Reserved for future use.</td>
</tr>
<tr>
<td>Number accepted connections</td>
<td>100</td>
<td>64</td>
<td>4</td>
<td>Binary</td>
<td>The number of accepted connections.</td>
</tr>
<tr>
<td>Number connections</td>
<td>104</td>
<td>68</td>
<td>4</td>
<td>Binary</td>
<td>Number of connections refused due to memory exceeded.</td>
</tr>
</tbody>
</table>
### Extended Section for the PowerExchange Client

The following table describes the fields in the extended section for the PowerExchange client:

<table>
<thead>
<tr>
<th>Field</th>
<th>Decimal Offset</th>
<th>Hexadecimal Offset</th>
<th>Length</th>
<th>Format</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>STCK start</td>
<td>0</td>
<td>0</td>
<td>8</td>
<td>Binary</td>
<td>Starting time (STCK).</td>
</tr>
<tr>
<td>STCK end</td>
<td>8</td>
<td>8</td>
<td>8</td>
<td>Binary</td>
<td>Ending time (STCK).</td>
</tr>
<tr>
<td>CPU time</td>
<td>16</td>
<td>10</td>
<td>8</td>
<td>Binary</td>
<td>CPU time of task, in microseconds.</td>
</tr>
<tr>
<td>Bytes sent</td>
<td>24</td>
<td>18</td>
<td>4</td>
<td>Binary</td>
<td>Number of bytes sent.</td>
</tr>
<tr>
<td>Messages sent</td>
<td>28</td>
<td>1C</td>
<td>4</td>
<td>Binary</td>
<td>Number of messages sent.</td>
</tr>
<tr>
<td>Bytes received</td>
<td>32</td>
<td>20</td>
<td>4</td>
<td>Binary</td>
<td>Number of bytes received.</td>
</tr>
<tr>
<td>Messages received</td>
<td>36</td>
<td>24</td>
<td>4</td>
<td>Binary</td>
<td>Number of messages received.</td>
</tr>
<tr>
<td>RC 1</td>
<td>40</td>
<td>28</td>
<td>4</td>
<td>Binary</td>
<td>Return code.</td>
</tr>
<tr>
<td>RC 2</td>
<td>44</td>
<td>2C</td>
<td>4</td>
<td>Binary</td>
<td>Return code.</td>
</tr>
<tr>
<td>RC 3</td>
<td>48</td>
<td>30</td>
<td>4</td>
<td>Binary</td>
<td>Return code.</td>
</tr>
</tbody>
</table>
### Extended Section for Data Access

The following table describes the fields in the extended section for data access:

<table>
<thead>
<tr>
<th>Field</th>
<th>Decimal Offset</th>
<th>Hexadecimal Offset</th>
<th>Length</th>
<th>Format</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>AM Type</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>Char</td>
<td>Access method type.</td>
</tr>
<tr>
<td>Reserved</td>
<td>1</td>
<td>1</td>
<td>7</td>
<td>Binary</td>
<td>Reserved for future use.</td>
</tr>
<tr>
<td>Rows Read</td>
<td>18</td>
<td>12</td>
<td>4</td>
<td>Binary</td>
<td>Number of rows read.</td>
</tr>
<tr>
<td>Bytes Read</td>
<td>22</td>
<td>16</td>
<td>8</td>
<td>Binary</td>
<td>Number of bytes read.</td>
</tr>
<tr>
<td>Rows Written</td>
<td>30</td>
<td>1E</td>
<td>4</td>
<td>Binary</td>
<td>Number of rows written.</td>
</tr>
<tr>
<td>Bytes Written</td>
<td>34</td>
<td>22</td>
<td>8</td>
<td>Binary</td>
<td>Number of bytes written.</td>
</tr>
<tr>
<td>Access method</td>
<td>42</td>
<td>2A</td>
<td>20</td>
<td>Char</td>
<td>The name of the access method.</td>
</tr>
</tbody>
</table>

### Function/Access Method Section (DB2)

The following table describes the fields in the function/access method section for DB2:

<table>
<thead>
<tr>
<th>Field</th>
<th>Decimal Offset</th>
<th>Decimal Offset</th>
<th>Length</th>
<th>Format</th>
<th>Value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>DB2 time</td>
<td>0</td>
<td>0</td>
<td>8</td>
<td>Binary</td>
<td>-</td>
<td>CPU time, in microseconds.</td>
</tr>
<tr>
<td>No. statements</td>
<td>8</td>
<td>8</td>
<td>4</td>
<td>Binary</td>
<td>-</td>
<td>The number of DB2 statements executed.</td>
</tr>
<tr>
<td>No. rows</td>
<td>12</td>
<td>1C</td>
<td>4</td>
<td>Binary</td>
<td>-</td>
<td>The number of rows returned.</td>
</tr>
<tr>
<td>SSID</td>
<td>16</td>
<td>10</td>
<td>4</td>
<td>Char</td>
<td>-</td>
<td>The DB2 SSID.</td>
</tr>
<tr>
<td>DB2 plan name</td>
<td>20</td>
<td>14</td>
<td>8</td>
<td>Char</td>
<td>-</td>
<td>The DB2 plan name.</td>
</tr>
<tr>
<td>Connection type</td>
<td>28</td>
<td>1C</td>
<td>8</td>
<td>Char</td>
<td>-</td>
<td>RRSAF/CAF</td>
</tr>
<tr>
<td>Authorization id</td>
<td>36</td>
<td>24</td>
<td>8</td>
<td>Char</td>
<td>-</td>
<td>The authorization ID.</td>
</tr>
<tr>
<td>Field</td>
<td>Decimal Offset</td>
<td>Decimal Offset</td>
<td>Length</td>
<td>Format</td>
<td>Value</td>
<td>Description</td>
</tr>
<tr>
<td>---------------</td>
<td>----------------</td>
<td>----------------</td>
<td>--------</td>
<td>--------</td>
<td>-------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
</tbody>
</table>
| Correlation id| 44             | 2C             | 32     | Char   | PWXnnnnn | Correlation ID, in the following format: PWX nnnnn  
Where nnnnn is a unique number that the PowerExchange Listener generates. PowerExchange generates a correlation ID for each subtask that the PowerExchange Listener generates. |
| SQL Code      | 76             | 4C             | 4      | Binary | -     | The SQL code.                                                               |
| Reason Code   | 80             | 50             | 4      | Binary | -     | The reason code.                                                            |
| Updated       | 84             | 54             | 4      | Binary | -     | The number of rows updated.                                                |
| Deleted       | 88             | 58             | 4      | Binary | -     | The number of rows deleted.                                                |
| Inserted      | 92             | 5C             | 4      | Binary | -     | The number of rows inserted.                                               |
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: DMRESOURCE, MVSDB2AF, and RACFCLASS

Required: No

Syntax:
\[
\text{SECURITY} = \{ 0 | 1 | 2 \}, \{ \text{N} | \text{Y} \}
\]

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, 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 form 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 form 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, PowerExchange does not require users to specify an operating system user 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.

**Usage Notes:**

- In the z/OS Installation Assistant, if you click **AdvancedParms** 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 DB22OS 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 MVS log files on z/OS, 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 199
- "Linux, UNIX, and Windows Security" on page 203
- "z/OS Security" on page 206
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 222. 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 199
- "User Authentication for the pwxcmd or infacmd pwx Programs on Linux, UNIX, and Windows" on page 206
- "Authorization to Run pwxcmd Commands on i5/OS" on page 202
- “PowerExchange Selective Sign-on” on page 224

### 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:

<table>
<thead>
<tr>
<th>Library and File Name</th>
<th>Authority Required</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>datalib/CFG</code></td>
<td>*USE</td>
</tr>
<tr>
<td><code>datalib/LOG</code></td>
<td>*CHANGE</td>
</tr>
<tr>
<td><code>datalib/Pnnnnn</code></td>
<td>*CHANGE</td>
</tr>
<tr>
<td><strong>Note</strong>: If you use PowerExchange alternative logging, the PowerExchange Listener creates a file named <em>Pnnnnn</em> in the data library, where <em>nnnnn</em> is the port number of the PowerExchange Listener.</td>
<td></td>
</tr>
<tr>
<td><code>dtllib/DTLMSG</code></td>
<td>*USE</td>
</tr>
<tr>
<td><code>dtllib/LICENSE</code></td>
<td>*USE</td>
</tr>
<tr>
<td><code>stdatamaps</code></td>
<td>*CHANGE</td>
</tr>
</tbody>
</table>

The following table lists the access authority that the PowerExchange Listener and PowerExchange Condense require for bulk data and CDC processing:

<table>
<thead>
<tr>
<th>File Name</th>
<th>PowerExchange Listener Authority</th>
<th>PowerExchange Condense Authority</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>cndlib/CFGCOND</code></td>
<td>*USE</td>
<td>*USE</td>
</tr>
<tr>
<td><code>cndlib/PWXJRNLCKP</code></td>
<td>*CHANGE</td>
<td>*CHANGE</td>
</tr>
<tr>
<td><code>cpxlib</code></td>
<td>*CHANGE</td>
<td>n/a</td>
</tr>
<tr>
<td><code>datalib/CCT</code></td>
<td>*CHANGE</td>
<td>*USE</td>
</tr>
<tr>
<td><code>datalib/CDCT</code></td>
<td>*USE</td>
<td>*CHANGE</td>
</tr>
</tbody>
</table>
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:

<table>
<thead>
<tr>
<th>Object</th>
<th>Authority</th>
</tr>
</thead>
<tbody>
<tr>
<td>Journal</td>
<td>*OBJEXIST</td>
</tr>
<tr>
<td>Journal library</td>
<td>*EXECUTE</td>
</tr>
<tr>
<td>Journal receivers</td>
<td>*USE</td>
</tr>
<tr>
<td>Library that contains journal receivers</td>
<td>*EXECUTE</td>
</tr>
<tr>
<td>Files</td>
<td>*USE</td>
</tr>
<tr>
<td>Library that contains files</td>
<td>*EXECUTE</td>
</tr>
</tbody>
</table>

---

**Note:** If you use PowerExchange alternative logging, the PowerExchange Listener creates a file named `Pnnnnn` in the data library, where `nnnnn` is the port number of the PowerExchange Listener.

**Note:** If you use PowerExchange alternative logging, the PowerExchange Condense job creates a file named `JOBnnnnn` in the data library, where `nnnnn` is the job number of the PowerExchange Condense job.

---

**Table:**

<table>
<thead>
<tr>
<th>File Name</th>
<th>PowerExchange Listener Authority</th>
<th>PowerExchange Condense Authority</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>datalib/CDEP</code></td>
<td>*CHANGE</td>
<td>*CHANGE</td>
</tr>
<tr>
<td><code>datalib/CFG</code></td>
<td>*USE</td>
<td>*USE</td>
</tr>
<tr>
<td><code>datalib/LOG</code></td>
<td>*CHANGE</td>
<td>*CHANGE</td>
</tr>
<tr>
<td><code>datalib/Pnnnnn</code></td>
<td>*CHANGE</td>
<td>n/a</td>
</tr>
<tr>
<td><code>datalib/JOBnnnnn</code></td>
<td>n/a</td>
<td>*CHANGE</td>
</tr>
<tr>
<td><code>dtllib/DTLMSG</code></td>
<td>*USE</td>
<td>*USE</td>
</tr>
<tr>
<td><code>dtllib/LICENSE</code></td>
<td>*USE</td>
<td>*USE</td>
</tr>
<tr>
<td><code>stdatamaps</code></td>
<td>*CHANGE</td>
<td>n/a</td>
</tr>
</tbody>
</table>
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 $datalib/AUTHTSKLST security object on that system.

The following table lists the access authority that the PowerExchange Listener requires to run LISTTASK and STOPTASK commands:

<table>
<thead>
<tr>
<th>PowerExchange Listener Command</th>
<th>Library and File Name</th>
<th>Authority Required</th>
</tr>
</thead>
<tbody>
<tr>
<td>LISTTASK</td>
<td>$datalib/AUTHTSKLST</td>
<td>*USE</td>
</tr>
<tr>
<td>STOPTASK</td>
<td>$datalib/AUTHTSKSTP</td>
<td>*USE</td>
</tr>
</tbody>
</table>

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 DBMOVER 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/LCLOSFRCE security object on that system.

The following table lists the pwxcmd PowerExchange Listener commands and corresponding security objects:

<table>
<thead>
<tr>
<th>pwxcmd Command</th>
<th>Security Object</th>
</tr>
</thead>
<tbody>
<tr>
<td>close</td>
<td>LCLOSE</td>
</tr>
<tr>
<td>closeforce</td>
<td>LCLOSFRCE</td>
</tr>
<tr>
<td>listtask</td>
<td>LLISSERTASK</td>
</tr>
<tr>
<td>stoptask</td>
<td>LSTOPTASK</td>
</tr>
</tbody>
</table>

The following table lists the pwxcmd PowerExchange Condense commands and corresponding security objects:

<table>
<thead>
<tr>
<th>pwxcmd Command</th>
<th>Security Object</th>
</tr>
</thead>
<tbody>
<tr>
<td>condense</td>
<td>CCONDENSE</td>
</tr>
<tr>
<td>displaystatus</td>
<td>CDSPSTATUS</td>
</tr>
<tr>
<td>fileswitch</td>
<td>CFILSWITCH</td>
</tr>
<tr>
<td>shutcond</td>
<td>CSHUTCOND</td>
</tr>
<tr>
<td>shutdown</td>
<td>CSHUTDOWN</td>
</tr>
</tbody>
</table>

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 202
- "PowerExchange Sign-on File" on page 224

**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.

**RELATED TOPICS:**

- "User Authentication for the pwxcmd Program on i5/OS" on page 202
- "Authorization to Run pwxcmd and infacmd pwx Commands on Linux, UNIX, and Windows" on page 204
- "PowerExchange Selective Sign-on" on page 224

**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 224.

**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:

```plaintext
AUTHGROUP=(*,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 221
- "PowerExchange Sign-on File" on page 224

**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, such as HP1 for HP-UX.

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:

```plaintext
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](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.
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.

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 222. 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, pwxccl, 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 221
- "Authorization to Run pwxcmd Commands on z/OS" on page 221
- "PowerExchange Selective Sign-on" on page 224

**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:

<table>
<thead>
<tr>
<th>DD Name</th>
<th>PowerExchange Listener Authority</th>
<th>Netport Jobs Authority</th>
</tr>
</thead>
<tbody>
<tr>
<td>STEPLIB</td>
<td>READ</td>
<td>READ</td>
</tr>
<tr>
<td>DATAMAP</td>
<td>UPDATE</td>
<td>READ</td>
</tr>
<tr>
<td>DTLCFG</td>
<td>READ</td>
<td>READ</td>
</tr>
<tr>
<td>DTLKEY</td>
<td>READ</td>
<td>READ</td>
</tr>
<tr>
<td>DTLLOG</td>
<td>UPDATE</td>
<td>UPDATE</td>
</tr>
<tr>
<td>DTLLOGnn</td>
<td>UPDATE</td>
<td>UPDATE</td>
</tr>
</tbody>
</table>
The following table lists the access authority that PowerExchange jobs and started tasks require for CDC processing:

<table>
<thead>
<tr>
<th>DD Name</th>
<th>PowerExchange Listener Authority</th>
<th>PowerExchange Agent Authority</th>
<th>PowerExchange Logger Authority</th>
<th>PowerExchange Condense Authority</th>
</tr>
</thead>
<tbody>
<tr>
<td>DTLMSG</td>
<td>READ</td>
<td></td>
<td>READ</td>
<td></td>
</tr>
<tr>
<td>DTLSGN</td>
<td>READ</td>
<td></td>
<td>READ</td>
<td></td>
</tr>
</tbody>
</table>

**Note:** Dynamically allocated SYSOUT data set.
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:

<table>
<thead>
<tr>
<th>Job or Started Task</th>
<th>Data Sets</th>
<th>Authority Required</th>
</tr>
</thead>
<tbody>
<tr>
<td>PowerExchange Agent</td>
<td>CCT data set, which contains capture registrations</td>
<td>UPDATE</td>
</tr>
<tr>
<td>PowerExchange Logger</td>
<td>PRILOG and SECLOG log data sets</td>
<td>UPDATE</td>
</tr>
<tr>
<td>PowerExchange Logger</td>
<td>Archive log data sets</td>
<td>ALTER</td>
</tr>
<tr>
<td>PowerExchange Condense</td>
<td>Checkpoint data sets</td>
<td>ALTER</td>
</tr>
<tr>
<td>PowerExchange Condense</td>
<td>Condense files</td>
<td>ALTER</td>
</tr>
<tr>
<td>IMS log-based ECCR</td>
<td>The RECON data sets that the IMS log-based ECCR processes</td>
<td>READ</td>
</tr>
<tr>
<td>IMS log-based ECCR</td>
<td>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</td>
<td>READ</td>
</tr>
</tbody>
</table>

**Related Topics:**
- “Data Set Level Security for Capture Registrations and Extraction Maps” on page 213

**Data Set Access Requirements for PowerExchange ECCR**

You must authorize access to the data sets used by PowerExchange ECCR to the user ID under which the ECCR run.

PowerExchange ECCR 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:

<table>
<thead>
<tr>
<th>DD Name</th>
<th>Adabas ECCR Authority</th>
<th>Adabas User Exit 2 Job Authority</th>
<th>DB2 ECCR Authority</th>
<th>IMS Log-Based ECCR Authority</th>
<th>IMS Synchronous ECCR Authority</th>
</tr>
</thead>
<tbody>
<tr>
<td>STEPLIB</td>
<td>READ</td>
<td>READ</td>
<td>READ</td>
<td>READ</td>
<td>READ</td>
</tr>
<tr>
<td>DATAMAP</td>
<td>READ</td>
<td>-</td>
<td>-</td>
<td>READ</td>
<td>-</td>
</tr>
<tr>
<td>DDASSOR1</td>
<td>UPDATE</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>DDCARD</td>
<td>READ</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>DDDATAR1</td>
<td>UPDATE</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>DDWORKR1</td>
<td>UPDATE</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>DTLADKSD</td>
<td>READ</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>DTLAMCPR</td>
<td>READ</td>
<td>-</td>
<td>READ</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>DTLCACDC</td>
<td>-</td>
<td>-</td>
<td>READ</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>DTLCACFG</td>
<td>READ</td>
<td>-</td>
<td>READ</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>DTLCACHEG</td>
<td>-</td>
<td>-</td>
<td>READ</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>DTLCCADA</td>
<td>-</td>
<td>UPDATE</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>DTLCCPLG</td>
<td>-</td>
<td>READ</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>DTLCFG</td>
<td>READ</td>
<td>READ</td>
<td>-</td>
<td>READ</td>
<td>-</td>
</tr>
<tr>
<td>DTLKEY</td>
<td>READ</td>
<td>READ</td>
<td>-</td>
<td>READ</td>
<td>-</td>
</tr>
<tr>
<td>DTLLOG</td>
<td>UPDATE</td>
<td>UPDATE</td>
<td>-</td>
<td>UPDATE</td>
<td>-</td>
</tr>
<tr>
<td>DTLMSG</td>
<td>READ</td>
<td>READ</td>
<td>-</td>
<td>READ</td>
<td>-</td>
</tr>
<tr>
<td>EDMPARMS</td>
<td>READ</td>
<td>-</td>
<td>READ</td>
<td>READ</td>
<td>READ</td>
</tr>
<tr>
<td>REPL2CTL</td>
<td>-</td>
<td>-</td>
<td>READ</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>REPL2OPT</td>
<td>-</td>
<td>-</td>
<td>READ</td>
<td>-</td>
<td>-</td>
</tr>
</tbody>
</table>
The following table lists the access authority that the batch VSAM, CICS/VSAM, and Datacom synchronous ECCRs require for data sets, by DD name:

<table>
<thead>
<tr>
<th>DD Name</th>
<th>Batch VSAM ECCR Authority</th>
<th>CICS/VSAM ECCR Authority</th>
<th>Datacom Change Collector Authority</th>
<th>Datacom Change Controller Authority</th>
<th>Datacom Log Feeder Authority</th>
</tr>
</thead>
<tbody>
<tr>
<td>STEPLIB</td>
<td>READ</td>
<td>READ</td>
<td>READ</td>
<td>READ</td>
<td>READ</td>
</tr>
<tr>
<td>DFHRPL</td>
<td>-</td>
<td>READ</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>DTLAMCPR</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>READ</td>
<td>-</td>
</tr>
<tr>
<td>DTLACCFG</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>READ</td>
<td>-</td>
</tr>
<tr>
<td>DTLCFG</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>READ</td>
<td>-</td>
</tr>
<tr>
<td>DTLLOG</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>UPDATE</td>
<td>-</td>
</tr>
<tr>
<td>DTLLOGnn</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>UPDATE</td>
<td>-</td>
</tr>
<tr>
<td>DTLMSG</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>READ</td>
<td>-</td>
</tr>
<tr>
<td>EDMPARMS</td>
<td>READ</td>
<td>READ</td>
<td>-</td>
<td>-</td>
<td>READ</td>
</tr>
</tbody>
</table>

The following table lists the access authority that the Datacom table-based and IDMS log-based ECCRs require for data sets, by DD name:

<table>
<thead>
<tr>
<th>DD Name</th>
<th>Datacom Table-Based</th>
<th>IDMS Log-Based</th>
</tr>
</thead>
<tbody>
<tr>
<td>STEPLIB</td>
<td>READ</td>
<td>READ</td>
</tr>
<tr>
<td>DTLAMCPR</td>
<td>READ</td>
<td>READ</td>
</tr>
<tr>
<td>DTLACCFG</td>
<td>READ</td>
<td>READ</td>
</tr>
<tr>
<td>DTLCFG</td>
<td>N/A</td>
<td>READ</td>
</tr>
<tr>
<td>DTLLOG</td>
<td>UPDATE</td>
<td>UPDATE</td>
</tr>
<tr>
<td>DTLLOGnn</td>
<td>UPDATE</td>
<td>UPDATE</td>
</tr>
<tr>
<td>DTLKEY</td>
<td>READ</td>
<td>READ</td>
</tr>
<tr>
<td>DTLMSG</td>
<td>READ</td>
<td>READ</td>
</tr>
<tr>
<td>EDMPARMS</td>
<td>READ</td>
<td>READ</td>
</tr>
</tbody>
</table>

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 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 213.

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 208
- "RACF Class Security for Capture Registrations and Extractions Maps" on page 213

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 128 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 UPDATE 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:

<table>
<thead>
<tr>
<th>Variable</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>dbtype</td>
<td>The database type, which is one of the following:</td>
</tr>
<tr>
<td></td>
<td>- ADA. Adabas.</td>
</tr>
<tr>
<td></td>
<td>- DB2. DB2.</td>
</tr>
<tr>
<td></td>
<td>- DCM. Datacom.</td>
</tr>
<tr>
<td></td>
<td>- IDM. IDMS.</td>
</tr>
<tr>
<td></td>
<td>- IMS. IMS.</td>
</tr>
<tr>
<td></td>
<td>- VSM. VSAM.</td>
</tr>
<tr>
<td>dbid</td>
<td>The instance, database ID, or collection ID specified in the registration group that contains the capture registration. Valid value is a character string.</td>
</tr>
<tr>
<td>registration_name</td>
<td>The name of the capture registration. Valid value is a character string.</td>
</tr>
</tbody>
</table>

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:

<table>
<thead>
<tr>
<th>Variable</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>dbid</td>
<td>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.</td>
</tr>
<tr>
<td>extraction_map_name</td>
<td>The name of the extraction map. Valid value is a character string.</td>
</tr>
</tbody>
</table>

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:

<table>
<thead>
<tr>
<th>Task</th>
<th>Required Data Set Access Authority</th>
<th>Required Resource Profile Access Authority</th>
</tr>
</thead>
<tbody>
<tr>
<td>View capture registrations.</td>
<td>UPDATE access to the data set assigned to the DTLAMCPR DD name in the PowerExchange Listener JCL</td>
<td>READ access to the CAPX.REG.* resource profile</td>
</tr>
<tr>
<td>Add capture registrations.</td>
<td>UPDATE access to the data set assigned to the DTLAMCPR DD name in the PowerExchange Listener JCL</td>
<td>UPDATE access to the CAPX.REG.* resource profile</td>
</tr>
<tr>
<td>Edit capture registrations.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Delete capture registrations.</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
The following table lists the required data set access authority and resource profile access authority for extraction maps:

<table>
<thead>
<tr>
<th>Task</th>
<th>Required Data Set Access Authority</th>
<th>Required Resource Profile Access Authority</th>
</tr>
</thead>
<tbody>
<tr>
<td>View extraction maps.</td>
<td>READ access to the data set assigned to the DTLCAMAP DD name in the PowerExchange Listener JCL</td>
<td>None</td>
</tr>
<tr>
<td>Extract change data.</td>
<td>READ access to the data set assigned to the DTLCAMAP DD name in the PowerExchange Listener JCL</td>
<td>READ access to the CAPX.CND.* resource profile</td>
</tr>
<tr>
<td>Add extraction maps.</td>
<td>- UPDATE access to the data set assigned to the DTLAMCPR DD name in the PowerExchange Listener JCL</td>
<td>READ access to the CAPX.REG.* resource profile</td>
</tr>
<tr>
<td></td>
<td>- UPDATE access to the data set assigned to the DTLCAMAP DD name in the PowerExchange Listener JCL</td>
<td></td>
</tr>
</tbody>
</table>

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 128 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 DMRESOURCE parameter in the DBMOVER configuration file, or if DMRESOURCE is not specified, the default of DATASET.

The following table describes the types of access authorities that you can include:

<table>
<thead>
<tr>
<th>Access Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>READ</td>
<td>Allows users to read data maps</td>
</tr>
<tr>
<td>UPDATE</td>
<td>Allows users to define, delete, and modify data maps</td>
</tr>
</tbody>
</table>

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 DMRESOURCE 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
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 128 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 128 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 zeroes 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 read access to Datacom tables. 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 128 bytes.

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 READ access. 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 or extraction 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 capture registration or 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
- RRSAF 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 RRSAF 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 128 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:

<table>
<thead>
<tr>
<th>Variable</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>service_type</td>
<td>PowerExchange service type. Options are:</td>
</tr>
<tr>
<td></td>
<td>- LISTENER. PowerExchange Listener.</td>
</tr>
<tr>
<td></td>
<td>- CONDENSE. PowerExchange Condense.</td>
</tr>
<tr>
<td>service_name</td>
<td>The service name specified in the CMDNODE statement in the DBMOVER configuration file.</td>
</tr>
<tr>
<td>command_name</td>
<td>The pwxcmd command. You can issue the following PowerExchange Condense commands through the pwxcmd program:</td>
</tr>
<tr>
<td></td>
<td>- condense</td>
</tr>
<tr>
<td></td>
<td>- displaystatus</td>
</tr>
<tr>
<td></td>
<td>- fileswitch</td>
</tr>
<tr>
<td></td>
<td>- shutcond</td>
</tr>
<tr>
<td></td>
<td>- shutdown</td>
</tr>
<tr>
<td></td>
<td>You can issue the following PowerExchange Listener commands through the pwxcmd program:</td>
</tr>
<tr>
<td></td>
<td>- close</td>
</tr>
<tr>
<td></td>
<td>- closeforce</td>
</tr>
<tr>
<td></td>
<td>- listtask</td>
</tr>
<tr>
<td></td>
<td>- stoptask</td>
</tr>
</tbody>
</table>

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:

```plaintext
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:

```plaintext
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 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 224

### 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]
[TASKCTRL={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:

<table>
<thead>
<tr>
<th>Operating System</th>
<th>SIGNON File Name and Location</th>
</tr>
</thead>
<tbody>
<tr>
<td>i5/OS</td>
<td>SIGNON member in the datalib /CFG file</td>
</tr>
<tr>
<td>Linux, UNIX, and Windows</td>
<td>signon.txt file in the PowerExchange installation directory</td>
</tr>
<tr>
<td>z/OS</td>
<td>SIGNON member in RUNLIB to which the DTLSGN DD in the PowerExchange Listener JCL points</td>
</tr>
</tbody>
</table>

**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:

<table>
<thead>
<tr>
<th>Command</th>
<th>Corresponds to infacmd pwx Command</th>
<th>Corresponds to pwxcmd Command</th>
</tr>
</thead>
<tbody>
<tr>
<td>CLOSE</td>
<td>CloseListener</td>
<td>close</td>
</tr>
<tr>
<td>CLOSEFORCE</td>
<td>CloseForceListener</td>
<td>closeforce</td>
</tr>
<tr>
<td>LISTTASK</td>
<td>ListTaskListener</td>
<td>listtask</td>
</tr>
<tr>
<td>STOPTASK</td>
<td>StopTaskListener</td>
<td>stoptask</td>
</tr>
</tbody>
</table>

The following table describes the commands that you can specify for PowerExchange Condense:

<table>
<thead>
<tr>
<th>Command</th>
<th>Corresponds to infacmd pwx Commands</th>
<th>Corresponds to pwxcmd Command</th>
</tr>
</thead>
<tbody>
<tr>
<td>CONDENSE</td>
<td>-</td>
<td>condense</td>
</tr>
<tr>
<td>DISPLAYSTATUS</td>
<td>-</td>
<td>displaystatus</td>
</tr>
<tr>
<td>FILESWITCH</td>
<td>-</td>
<td>fileswitch</td>
</tr>
<tr>
<td>SHUTCOND</td>
<td>-</td>
<td>shutcond</td>
</tr>
<tr>
<td>SHUTDOWN</td>
<td>-</td>
<td>shutdown</td>
</tr>
</tbody>
</table>
The following table describes the commands that you can specify for the PowerExchange Logger for Linux, UNIX, and Windows:

<table>
<thead>
<tr>
<th>Command</th>
<th>Corresponds to infacmd pwx Command</th>
</tr>
</thead>
<tbody>
<tr>
<td>CONDENSE</td>
<td>CondenseLogger</td>
</tr>
<tr>
<td>DISPLAYALL</td>
<td>DisplayAllLogger</td>
</tr>
<tr>
<td>DISPLAYCHECKPOINTS</td>
<td>DisplayCheckpointsLogger</td>
</tr>
<tr>
<td>DISPLAYCPU</td>
<td>DisplayCPUTLogger</td>
</tr>
<tr>
<td>DISPLAYEVENTS</td>
<td>DisplayEventsLogger</td>
</tr>
<tr>
<td>DISPLAYMEMORY</td>
<td>DisplayMemoryLogger</td>
</tr>
<tr>
<td>DISPLAYRECORDS</td>
<td>DisplayRecordsLogger</td>
</tr>
<tr>
<td>DISPLAYSTATUS</td>
<td>DisplayStatusLogger</td>
</tr>
<tr>
<td>FILESWITCH</td>
<td>FileSwitchLogger</td>
</tr>
<tr>
<td>SHUTCOND</td>
<td>-</td>
</tr>
<tr>
<td>SHUTDOWN</td>
<td>ShutDownLogger</td>
</tr>
</tbody>
</table>

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:

```plaintext
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:

```plaintext
USER={user_ID[@domain]}
ALLOW={F|N|Y}
[AUTHGROUP={group_id}]
[IP=nnnn.nnnn.nnnn.nnnn]
[TASKCTRL={D|S}]
```
The USER statement has the following parameters:

```plaintext
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 an 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.

```plaintext
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.

```plaintext
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.

```plaintext
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:

```plaintext
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 coverts 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_CNTL** 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=( ALL)
```

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 ALL)
USER=( ALL)
```

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)
```
Secure Sockets Layer Support

This chapter includes the following topics:

- SSL Security on a PowerExchange Network, 231
- PowerExchange SSL Architecture, 232
- FIPS 140-2 Compliance, 232
- PWXUSSL Utility, 234
- PowerExchange SSL Configuration Steps, 234
- SSL Server Configuration on z/OS, 235
- SSL Client and Server Configuration on Linux, UNIX, and Windows, 239
- Verifying Connections, 244

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 TTLSCipherParms 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 TTLSCipherParms 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:

<table>
<thead>
<tr>
<th>OpenSSL Cipher Suite Name</th>
<th>AT-TLS Cipher Suite Name</th>
<th>Hexadecimal Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>DHE-RSA-AES256-SHA</td>
<td>TLS_DHE_RSA_WITH_AES_256_CBC_SHA</td>
<td>39</td>
</tr>
<tr>
<td>DHE-DSS-AES256-SHA</td>
<td>TLS_DHE_DSS_WITH_AES_256_CBC_SHA</td>
<td>38</td>
</tr>
<tr>
<td>AES256-SHA</td>
<td>TLS_RSA_WITH_AES_256_CBC_SHA</td>
<td>35</td>
</tr>
<tr>
<td>EDH-RSA-DES-CBC3-SHA</td>
<td>TLS_DHE_RSA_WITH_3DES_EDE_CBC_SHA</td>
<td>16</td>
</tr>
<tr>
<td>EDH-DSS-DES-CBC3-SHA</td>
<td>TLS_DHE_DSS_WITH_3DES_EDE_CBC_SHA</td>
<td>13</td>
</tr>
<tr>
<td>DES-CBC3-SHA</td>
<td>TLS_RSA_WITH_3DES_EDE_CBC_SHA</td>
<td>0A</td>
</tr>
<tr>
<td>DHE-RSA-AES128-SHA</td>
<td>TLS_DHE_RSA_WITH_AES_128_CBC_SHA</td>
<td>33</td>
</tr>
<tr>
<td>DHE-DSS-AES128-SHA</td>
<td>TLS_DHE_DSS_WITH_AES_128_CBC_SHA</td>
<td>32</td>
</tr>
<tr>
<td>AES128-SHA</td>
<td>TLS_RSA_WITH_AES_128_CBC_SHA</td>
<td>2F</td>
</tr>
</tbody>
</table>

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. For more information about PWXUSSL, 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

PWXUSSL Utility

Use the PowerExchange PWXUSSL utility to generate reports about SSL libraries and certificates on Linux, UNIX, and Windows.

You can generate the following reports:

- Version report. Reports the version of OpenSSL that was used to build the cryptographic library.
- Ciphers report. Reports the cipher suites that are available in the OpenSSL cryptographic library.

For more information about PWXUSSL, see the PowerExchange Utilities Guide.

PowerExchange SSL Configuration Steps

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 client and server.
   a. Create a CA certificate.
   b. Create a personal certificate.
3. Make the certificates available to clients or servers that require authentication.
4. Verify connections.

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.

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 specifies the z/OS jobs that are authorized to use SSL. Any job that communicates with an SSL Listener must also be authorized in the policy file. 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.

IBM provides the Network Configuration Assistant tool to assist in building an AT-TLS configuration and policy files. This is a graphical user interface that can be downloaded from the IBM z/OS support web site.

Get SSL certificates for all machines that use SSL connection in the PowerExchange environment. The system administrator will supply these certificates.

Related Topics:

- "Example AT-TLS Policy File for PowerExchange" on page 237

Adding an AT-TLS Rule

To add a rule, edit the policy file or use the IBM Configuration Assistant for z/OS Communications Server.

You can download the IBM Configuration Assistant from the IBM z/OS support web site.
The following table shows the statements to include when you add a rule:

<table>
<thead>
<tr>
<th>Statement</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>LocalPortRange</td>
<td>PowerExchange Listener port number.</td>
</tr>
<tr>
<td>Jobname</td>
<td>PowerExchange Listener job name.</td>
</tr>
<tr>
<td>Direction</td>
<td>Direction of communication. Specify <strong>Inbound</strong> to indicate that</td>
</tr>
<tr>
<td></td>
<td>communication proceeds from client to Listener.</td>
</tr>
<tr>
<td>TTLSGroupActionRef</td>
<td>References an existing <strong>group_action</strong> that is defined in another</td>
</tr>
<tr>
<td></td>
<td>section of the policy file.</td>
</tr>
<tr>
<td>TTLEnvironmentActionRef</td>
<td>References an existing <strong>environment_action</strong> that is defined in</td>
</tr>
<tr>
<td></td>
<td>another section of the policy file.</td>
</tr>
</tbody>
</table>

**Example Rule**

The following statements show an example rule:

```plaintext
TTLSRULE JOB_JBBV861
{
  LocalPortRange   13132
  Jobname          JBBV861
  Direction        Inbound
  TTLSGroupActionRef gActEnableTTLSEnvironmentActionRef eActServerDefault
}
```

**References to Existing Sections of Policy File**

The following table describes the existing sections of the policy file that the rule references:

<table>
<thead>
<tr>
<th>Statement</th>
<th>Sub-Statement</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>TTLS Group Action</td>
<td>TTLSEnabled</td>
<td>On</td>
</tr>
<tr>
<td>TTLS Group Action</td>
<td>CtraceClearText</td>
<td>Off</td>
</tr>
<tr>
<td>TTLS Group Action</td>
<td>Trace</td>
<td>7</td>
</tr>
<tr>
<td>TTLSEnvironmentAction</td>
<td>HandshakeRole</td>
<td>For servers, specifies one of the following values:</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- <strong>Server</strong>. The Listener act as the SSL server and does not</td>
</tr>
<tr>
<td></td>
<td></td>
<td>require client authentication.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- <strong>ServerWithClientAuth</strong>. The Listener act as the SSL server</td>
</tr>
<tr>
<td></td>
<td></td>
<td>and requires client authentication.</td>
</tr>
<tr>
<td>TTLSEnvironmentAction</td>
<td>TTLSCipherParmsRef</td>
<td>References the TTLSCipherParms statement.</td>
</tr>
<tr>
<td>TTLSEnvironmentAction</td>
<td>TLSKeyRingParmsRef</td>
<td>References the TLSKeyRingParms statement.</td>
</tr>
<tr>
<td>TTLSCipherParms</td>
<td>V3CipherSuites</td>
<td>Supported symmetric cipher suites.</td>
</tr>
<tr>
<td>TLSKeyRingParms</td>
<td>Keyring</td>
<td>Key ring that contains the personal and CA certificates.</td>
</tr>
</tbody>
</table>

The following statements show example sections of a referenced policy file:

```plaintext
TTLSGroupAction  gActEnableTTLSEnvironmentActionRef
{
```
Example AT-TLS Policy File for PowerExchange

The following example shows a complete AT-TLS policy file for PowerExchange:

```plaintext
# AT-TLS Policy Agent Configuration file for:
#  Image: MHEZ
# Stack: TCP/IP
# 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.
#
# TLRule 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 key8-MHEZ TTSKeyRingParms statement.
# - Connection Action cAct1 specifies that encryption parameters defined by
#   TTSCipherParms statement cipher1-AT-TLS_Silver, and that the listener is to act
#   as SSL server (without client certificates).
#
# TLRule 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).
#
# TTSCipherParms cipher1-AT-TLS_Silver shows a list on Cipher Suites which can be
# used
# for the connections.
#
# TLRule 0~1
#
# LocalAddrSetRef addr1
# RemoteAddrSetRef addr1
# Jobname PWXSSL85
# Direction Inbound
# Priority 255
# TTLSGroupActionRef gAct1
# TTSEnvironmentActionRef eAct1
# TTSConnectionActionRef cAct1
#]
# TLRule 0~5
#
# LocalAddrSetRef addr1
# RemoteAddrSetRef addr1
```

TLRule 0~1

- LocalAddrSetRef addr1
- RemoteAddrSetRef addr1
- Jobname PWXSSL85
- Direction Inbound
- Priority 255
- TTLSGroupActionRef gAct1
- TTSEnvironmentActionRef eAct1
- TTSConnectionActionRef cAct1

TLRule 0~5

- LocalAddrSetRef addr1
- RemoteAddrSetRef addr1
Create a key ring.

Create a personal certificate.

Creating a Personal Certificate on z/OS

To create a personal 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:

```plaintext
/* Create a Keyring for the application */
RACCCERT ID(MYUSERID) ADDRING(ATTLS_keyring)
SETROPTS RACLIST(DIGTCERT,DIGTNMAP) REFRESH

/* Create a certificate for the Server application */
RACCCERT ID(MYUSERID) GENCERT -
SUBJECTDN { -
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. */
RACCCERT 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 */
RACCCERT 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:

```plaintext
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.
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 pmlibeay32.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. PowerExchange works with any version of OpenSSL.

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.
   -sha1
   Specifies to use the SHA-1 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_CAE -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.

-sha1
  Specifies to use the SHA-1 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 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 personalreq.pem
   -newkey rsa:2048
   Requests a new certificate request and a 2048-bit RSA private key.

   -sha1
   Specifies to use the SHA-1 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 personalreq.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.

   -sha1
   Specifies to use the SHA-1 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 personal 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.

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 109

**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|ZOSSSL}
```

Use the SSL parameter to access a Linux, UNIX, or Windows system.

Use the ZOSSSL parameter to access a z/OS system. However, use the SSL parameter, instead of the ZOSSSL 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 ZOSSSL parameter. Five empty parameters appear between the port number parameter and the SSL or ZOSSSL 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 requires server authentication. When you configure an SSL client to require 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:

\[
\text{SSL_REQ_SRVR_CERT} = \{N\mid Y\}
\]

When the client requires certification of server certificates, the SSL_ALLOW_SELFSSIGNED statement specifies whether a self-signed certificate is sufficient to authenticate the server. Use the following syntax:

\[
\text{SSL_ALLOW_SELFSSIGNED} = \{N\mid Y\}
\]

If you configure the client to require 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 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:

- "NODE Statement" on page 120

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:

\[
\text{DTLREXE} \text{ PROG=PING LOC=sslnode}
\]

If the connection is working, the Listener issues the following message:

\[
\text{PWX=00755 DTLREXE Command OK!}
\]
Chapter 8

PowerExchange Alternative Network Security

This chapter includes the following topics:

- PowerExchange Alternative Network Security Overview, 245
- Modes of PowerExchange Encryption, 245
- Setting Encryption Defaults in the DBMOVER Configuration File, 247

Related Topics:

- "ENCRYPT Statement" on page 97

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 DES and RSA Security RC2 encryption.

To enable alternative network security, define ENCRYPT and ENCRYPTLEVEL statements in the DBMOVER configuration file on the system where the data movement process runs. 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
• DES encryption
• RC2 encryption

The RC2 and DES encryption types are not supported on 64-bit Windows or Linux on IBM System z.

PowerExchange Encryption Standard
This is a PowerExchange proprietary encryption algorithm.

DES Encryption Standard
The Digital Encryption Standard (DES) is a U.S. commercial encryption standard that has been available for over 15 years. The federal standard document FIPS PUB 46-2 describes this algorithm. The key consists of 56 random bits and 8 parity bits, forming a 64-bit or 8-byte key.

Triple DES Encryption Standard
Triple DES executes DES three times, which triples the number of bits in an encryption key.

A number of different methods achieve this function. This technique is known as Encrypt-Decrypt-Encrypt (EDE). The decryption process in the middle stage of Triple DES encryption provides compatibility with DES. If the three keys are the same, the Triple DES operation is equivalent to a single DES encryption. That way, an application that has only DES capabilities can still communicate with applications that use Triple DES. If the three keys are different, the decryption in the middle will scramble the message further. It will not decrypt the first stage. Triple DES decryption is the inverse operation of the previous sequence, that is, DES decryption followed by DES encryption and then another DES decryption.

The following figure illustrates Triple DES encryption:

RC2 Cipher
The RC2 cipher was developed by Ronald Rivest as an alternative to DES encryption. It is proprietary to RSA Security. The RC2 input key can be of any length from 1 to 128 bytes. The algorithm uses the input key to generate an effective key that is actually used for encryption purposes. Internally, the algorithm builds a key table based on the bits of the key data. The chosen number of effective key bits limits the number of possible key tables. The effective key size is variable and takes values from 1 bit up to 1024 bits.
The RC2 cipher is a variable-key-size block cipher. Though a DES key requires exactly eight bytes, an RC2 key can vary between one and 128 bytes. The larger the key, the greater the security. The RC2 cipher is called a block cipher because it encrypts 8-byte blocks.

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 Services or Data Quality mapping, you must 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.
This chapter includes the following topics:

- **PowerExchange Nonrelational SQL Overview, 248**
- **PowerExchange Nonrelational SQL Syntax, 248**
- **DTLDESCRIBE Metadata Syntax, 252**
- **DTLDESCRIBE Qualifiers, 268**
- **Restrictions on PowerExchange Nonrelational SQL, 270**

**PowerExchange Nonrelational SQL Overview**

To access nonrelational sources, such as an IMS database or VSAM file, use the PowerExchange Navigator to define 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.

**PowerExchange Nonrelational SQL Syntax**

The basic syntax supported by nonrelational data sources is as follows:

```sql
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

A basic comparison is where a column is compared with either another column or a constant.
Complex expressions such as (col1 + col2) or (col1 * 3) are not supported. For example:
col1 = ‘123’
The following table lists the valid operators:

<table>
<thead>
<tr>
<th>Name</th>
<th>Operator</th>
</tr>
</thead>
<tbody>
<tr>
<td>=</td>
<td>Equals</td>
</tr>
<tr>
<td>&lt;&gt;</td>
<td>Not equal</td>
</tr>
<tr>
<td>!=</td>
<td>Not equal</td>
</tr>
<tr>
<td>&gt;</td>
<td>Greater than</td>
</tr>
<tr>
<td>&gt;=</td>
<td>Not less than</td>
</tr>
<tr>
<td>&lt;</td>
<td>Less than</td>
</tr>
<tr>
<td>&lt;=</td>
<td>Not greater than</td>
</tr>
</tbody>
</table>

• Character constants must be enclosed in single quotation marks.
• 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.

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**

\[
YYYY-MM-DD\;HH:MI:SS.NNNNN\n\]

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:

```sql
coll [NOT] LIKE pattern
```

The pattern is a character constant 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:

<table>
<thead>
<tr>
<th>Pattern</th>
<th>Matches</th>
<th>Does Not Match</th>
</tr>
</thead>
<tbody>
<tr>
<td>%TAB%</td>
<td>TAB, ATAB, TABLE, TABULAR, MY_TABLE</td>
<td>MY_TAABLE</td>
</tr>
<tr>
<td>TA%</td>
<td>TAB, TABLE, TABULAR</td>
<td>ATAB, MY_TABLE, MY_TAABLE</td>
</tr>
<tr>
<td>TA_LE</td>
<td>TABLE</td>
<td>TAB, ATAB, TABULAR, MY_TABLE, MY_TAABLE</td>
</tr>
</tbody>
</table>

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`. 

---

250       Chapter 9: PowerExchange Nonrelational SQL
BETWEEN Comparison

This comparison is testing a column for being falling in side or outside (used with NOT) a range of values.

Example:

\[ \text{coll} \ [\text{NOT}] \ \text{BETWEEN} \ \text{valueA AND valueB} \]

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:

\[ \text{[NOT]} \ \{\text{coll} \ >\ = \ \text{valueA AND coll} \ <\ = \ \text{valueB} \} \]

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:

\[ \text{coll IS } [\text{NOT}] \ \text{NULL} \]

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:

\[
\text{Select * from IMSSC1.IMSMP1.IMSDemos Where KEY} \ >= 'A' \ \text{and KEY} \ <= 'D'
\]

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.IMSM1.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 (>='B' and <='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:

```
dtdescribe tables, [qualifier1], [qualifier2], [tablename], [comments], [ORDER],
[escape_character], [respectcase1], [respectcase2], [respectcase3],
[extended_information], [access_method]
```

After the basic syntax of dtdescribe tables, all the other parameters in brackets [ ] are optional. If any are coded, the intervening commas must be included so that the parser can determine which field is included.

The following table describes the parameters:

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>qualifier1</td>
<td>Returns the first and second data map qualifiers.</td>
</tr>
<tr>
<td>qualifier2</td>
<td></td>
</tr>
<tr>
<td>tablename</td>
<td>This is the primary table qualifier. Returns the table name. All tables are returned if left blank.</td>
</tr>
<tr>
<td>comments</td>
<td>If set to Y, returns any comments data found in the database.</td>
</tr>
<tr>
<td>ORDER</td>
<td>Leave this field blank. The comma is required to maintain position of subsequent fields.</td>
</tr>
<tr>
<td>escape_character</td>
<td>Specifies the character to be used as the escape character (default is ~).</td>
</tr>
<tr>
<td>respect case1</td>
<td>Set to Y this will respect the case of qualifier1, qualifier2 and tablename respectively - alternative is blank.</td>
</tr>
<tr>
<td>respect case2</td>
<td></td>
</tr>
<tr>
<td>respect case3</td>
<td></td>
</tr>
</tbody>
</table>
**Parameter**

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>extended_information</td>
<td>If set to Y, returns the following fields containing extra information. Applicable to NRDB, NRDB2, and CAPX only. Defaults is N (no extended information required).</td>
</tr>
</tbody>
</table>
| access_method       | 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 methods.  
  - ADABAS - A  
  - CAPX / RT - C  
  - DATACOM - X  
  - DB2 - Z  
  - DB2390IMG - G  
  - DB2UNLD - W  
  - DL1 - D  
  - ESDS - E  
  - IDMS - I  
  - ISAM - M  
  - KSDS - K  
  - MQ - Q  
  - ODBA - O  
  - RRDS - N  
  - SEQ - S  
  - TAPE - T  
  - USER - U  
  The following extraction map access methods are applicable only to CAPX and CAPXRT:  
  - XDB2 - B  
  - XMSSQL - L  
  - XORACLE - P |

**Related Topics:**

- [“DTLDESCRIBE Qualifiers” on page 268](#)

**DTLDESCRIBE TABLES Returned Information**

The following table describes the columns:

<table>
<thead>
<tr>
<th>Column Name</th>
<th>Description</th>
<th>Extended information</th>
</tr>
</thead>
<tbody>
<tr>
<td>qualifier_1</td>
<td>Creator/Schema name.</td>
<td>No</td>
</tr>
<tr>
<td>qualifier_2</td>
<td>PowerExchange Data Map name will be returned, or null for NRDB2.</td>
<td>No</td>
</tr>
<tr>
<td>Column Name</td>
<td>Description</td>
<td>Extended information</td>
</tr>
<tr>
<td>-------------</td>
<td>-------------</td>
<td>----------------------</td>
</tr>
<tr>
<td>tablename</td>
<td>Return table name.</td>
<td>No</td>
</tr>
<tr>
<td>comments</td>
<td>Comments column always present. Comment information is included only if requested by specifying the comments indicator in the describe.</td>
<td>No</td>
</tr>
<tr>
<td>type</td>
<td>Type of object for which metadata was returned, such as TABLE or MATERIALIZED VIEW.</td>
<td>No</td>
</tr>
<tr>
<td>Acc_mth</td>
<td>The access method (from data map)</td>
<td>Yes</td>
</tr>
<tr>
<td>Acc_mths01</td>
<td>Adabas</td>
<td>Database ID</td>
</tr>
<tr>
<td>Acc_mths01</td>
<td>SEQ, KSDS, ESDS</td>
<td>MapFileName</td>
</tr>
<tr>
<td>Acc_mths01</td>
<td>IDMS</td>
<td>MapSubSchemaName</td>
</tr>
<tr>
<td>Acc_mths01</td>
<td>ODBA or DL/I</td>
<td>DBDName</td>
</tr>
<tr>
<td>Acc_mths01</td>
<td>Datacom</td>
<td>Database ID</td>
</tr>
<tr>
<td>Acc_mths01</td>
<td>CAPX or CAPXRT</td>
<td>The access method of the original data map</td>
</tr>
<tr>
<td>Acc_mths01</td>
<td>Other access methods</td>
<td>NULL</td>
</tr>
<tr>
<td>Acc_mths02</td>
<td>Adabas</td>
<td>File Number</td>
</tr>
<tr>
<td>Acc_mths02</td>
<td>ODBA or DL/I</td>
<td>DBDType</td>
</tr>
<tr>
<td>Acc_mths02</td>
<td>Datacom</td>
<td>Table record length</td>
</tr>
<tr>
<td>Acc_mths02</td>
<td>IDMS or VSAM</td>
<td>C (if table is valid for data capture) or NULL</td>
</tr>
<tr>
<td>Acc_mths02</td>
<td>CAPX</td>
<td>Condense option (Full, Part or None)</td>
</tr>
<tr>
<td>Acc_mths02</td>
<td>DB2UNLD</td>
<td>Null indicator value</td>
</tr>
<tr>
<td>Acc_mths02</td>
<td>Other access methods</td>
<td>NULL</td>
</tr>
<tr>
<td>Acc_mths03</td>
<td>Datacom</td>
<td>Table ID</td>
</tr>
<tr>
<td>Acc_mths03</td>
<td>IDMS or VSAM</td>
<td>Database name</td>
</tr>
<tr>
<td>Acc_mths03</td>
<td>CAPX or CAPXRT</td>
<td>Creator of the capture registration</td>
</tr>
<tr>
<td>Acc_mths03</td>
<td>DB2</td>
<td>DB2 Sub system ID</td>
</tr>
<tr>
<td>Acc_mths03</td>
<td>DB2UNLD</td>
<td>Unload type</td>
</tr>
</tbody>
</table>
### Column Name | Description | Extended information
--- | --- | ---
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)

**Related Topics:**
- “DTLDESCRIBE Qualifiers” on page 268

**DTLDESCRIBE PROCEDURES Qualifier**

Enables you to list the available stored procedures:

```
dtldescribe procedures, [qualifier1], [qualifier2], [procedurename],
[escape_character], [respectcase1], [respectcase2], [respectcaseprocmname]
```

After the basic syntax of DTLDESCRIBE procedures all the other parameters [] are optional. If any are coded the intervening commas must be included so that the parser can determine which field is included.

The following table describes the parameters:

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>qualifier1</td>
<td>Catalog name.</td>
</tr>
<tr>
<td>qualifier2</td>
<td>Schema name.</td>
</tr>
<tr>
<td>procedurename</td>
<td>Procedure name</td>
</tr>
<tr>
<td>escape_character</td>
<td>Specifies the character to be used as the escape character. Default is ~.</td>
</tr>
</tbody>
</table>
### Parameter Description

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>respect case1</td>
<td>Set to Y this will respect the case of qualifier1 / qualifier2 - alternative is blank.</td>
</tr>
<tr>
<td>respect case2</td>
<td>Set to Y this will respect the case of the procedure name - alternative is blank.</td>
</tr>
</tbody>
</table>

### DTLDESCRIBE PROCEDURES Returned Information

The following table describes the columns:

<table>
<thead>
<tr>
<th>Column Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>PROCEDURE_CAT</td>
<td>Procedure Catalog, null if not present; VARCHAR.</td>
</tr>
<tr>
<td>PROCEDURE_SCHEM</td>
<td>Procedure Schema, null if not present; VARCHAR.</td>
</tr>
<tr>
<td>PROCEDURE_NAME</td>
<td>Procedure Name, not null; VARCHAR.</td>
</tr>
<tr>
<td>NUM_INPUT_PARAMS</td>
<td>Number of input parameters, not null (N/A for SQL Server); INTEGER.</td>
</tr>
<tr>
<td>NUM_OUTPUT_PARAMS</td>
<td>Number of output parameters, not null (N/A for SQL Server); INTEGER.</td>
</tr>
<tr>
<td>NUM_RESULT_SETS</td>
<td>Number of result sets, not null (N/A for SQL Server); INTEGER.</td>
</tr>
<tr>
<td>REMARKS</td>
<td>Comments; VARCHAR.</td>
</tr>
<tr>
<td>PROCEDURE_TYPE</td>
<td>Procedure type.</td>
</tr>
</tbody>
</table>

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], [procedurename], [columnname], [escape_character], [respectcase1], [respectcase2], [respectcaseprocname], [respectcasecolname]
```

After the basic syntax of DTLDESCRIBE procedurecolumns all the other parameters [] are optional. If any are coded the intervening commas must be included so that the parser can determine which field is included.

The following table describes the parameters:

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>qualifier1</td>
<td>Catalog name.</td>
</tr>
<tr>
<td>qualifier2</td>
<td>Schema name.</td>
</tr>
<tr>
<td>procedurename</td>
<td>Procedure name.</td>
</tr>
<tr>
<td>columnname</td>
<td>Column name.</td>
</tr>
<tr>
<td>Parameter</td>
<td>Description</td>
</tr>
<tr>
<td>------------------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>escape_character</td>
<td>Specifies the character to be used as the escape character. Default is ~.</td>
</tr>
<tr>
<td>respect case1</td>
<td>Set to Y to respect the case of qualifier1 / qualifier2 - alternative is blank.</td>
</tr>
<tr>
<td>respect case2</td>
<td>Set to Y this will respect the case of procedure name - alternative is blank.</td>
</tr>
<tr>
<td>respect caseprocname</td>
<td>Set to Y this will respect the case of column name - alternative is blank.</td>
</tr>
</tbody>
</table>

**DTLDESCRIBE PROCEDURECOLUMNS Returned Information**

The following table describes the columns:

<table>
<thead>
<tr>
<th>Column Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>PROCEDURE_CAT</td>
<td>Procedure catalog, null if not present; VARCHAR</td>
</tr>
<tr>
<td>PROCEDURE_SCHEM</td>
<td>Procedure schema, null if not present; VARCHAR</td>
</tr>
<tr>
<td>PROCEDURE_NAME</td>
<td>Procedure name; VARCHAR</td>
</tr>
<tr>
<td>COLUMN_NAME</td>
<td>Column name; VARCHAR</td>
</tr>
<tr>
<td>COLUMN_TYPE</td>
<td>Column type, not null; SMALLINT</td>
</tr>
<tr>
<td></td>
<td>Can be</td>
</tr>
<tr>
<td></td>
<td>0 - SQL_PARAM_TYPE_UNKNOWN</td>
</tr>
<tr>
<td></td>
<td>1 - SQL_PARAM_INPUT - input parameter</td>
</tr>
<tr>
<td></td>
<td>2 - SQL_PARAM_INPUT_OUTPUT - input/output parameter</td>
</tr>
<tr>
<td></td>
<td>3 - SQL_RESULT_COLUMN - Parm is a column inset</td>
</tr>
<tr>
<td></td>
<td>4 - SQL_PARAM_OUTPUT - output parameter</td>
</tr>
<tr>
<td></td>
<td>5 - SQL_RETURN_VALUE - Column is return value of procedure</td>
</tr>
<tr>
<td>DATA_TYPE</td>
<td>SQL datatype, not null; SMALLINT</td>
</tr>
<tr>
<td>TYPE_NAME</td>
<td>Type name, character string representing datatype, not null; VARCHAR</td>
</tr>
<tr>
<td>COLUMN_SIZE</td>
<td>Column size; INTEGER</td>
</tr>
<tr>
<td>BUFFER_LENGTH</td>
<td>Buffer length. Maximum number of bytes required to store column data; INTEGER</td>
</tr>
<tr>
<td>DECIMAL_DIGITS</td>
<td>Scale, NULL if not applicable; SMALLINT</td>
</tr>
<tr>
<td>NUM_PREC_RADIX</td>
<td>Precision; SMALLINT</td>
</tr>
<tr>
<td>NULLABLE</td>
<td>Nullable, not null, determines if column will accept a null value; SMALLINT.</td>
</tr>
<tr>
<td></td>
<td>Can be</td>
</tr>
<tr>
<td></td>
<td>0 - SQL_NO_NULLS</td>
</tr>
<tr>
<td></td>
<td>1 - SQL_NULLABLE</td>
</tr>
</tbody>
</table>
### Column Name

<table>
<thead>
<tr>
<th>Column Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>REMARKS</td>
<td>Remarks; VARCHAR</td>
</tr>
<tr>
<td>COLUMN_DEF</td>
<td>Column default value; VARCHAR</td>
</tr>
<tr>
<td>SQL_DATA_TYPE</td>
<td>SQL Datatype, not null; SMALLINT</td>
</tr>
<tr>
<td>SQL_DATETIME_SUB</td>
<td>Datetime subtype. Can be 1 - SQL_CODE_DATE, 2 - SQL_CODE_TIME, 3 - SQL_CODE_TIMESTAMP</td>
</tr>
<tr>
<td>CHAR_OCTET_LENGTH</td>
<td>Maximum length in bytes of a character type column; INTEGER</td>
</tr>
<tr>
<td>ORDINAL_POSITION</td>
<td>Ordinal Position if parameter given by Column name in result set, not null; INTEGER</td>
</tr>
<tr>
<td>IS_NULLABLE</td>
<td>Can the column contain nulls. “YES” if column can contain nulls, “NO” if not; VARCHAR</td>
</tr>
</tbody>
</table>

### DTLDESCRIBE PROCEDURECOLUMNS is implemented for DB2400C and MSSQL.

**DTLDESCRIBE COLUMNS Qualifier**

Enables you to list the available columns:

```sql
dtldescribe columns, [qualifier1], [qualifier2], [tablename],
[comments], [ORDER], [escape_character], [respectcase1],
[respectcase2], [respectcase3], [extended_information], [access_method]
```

After the basic syntax of dtldescribe columns all the other parameters [] are optional. If any are coded, the intervening commas must be included so that the parser can determine which field is included.

The following table describes the parameters:

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>qualifier1</td>
<td>-</td>
</tr>
<tr>
<td>qualifier2</td>
<td>Always NULL if describe is done using NRDB2 otherwise PowerExchange Data Map name.</td>
</tr>
<tr>
<td>tablename</td>
<td>This is the primary table qualifier. Returns the table name. Columns for all available tables are returned if left blank.</td>
</tr>
<tr>
<td>comments</td>
<td>If set to Y this will return any comments data found in the database.</td>
</tr>
<tr>
<td>ORDER</td>
<td>Optional argument, Y will force order by column number or blank will leave column order undefined.</td>
</tr>
<tr>
<td>escape_character</td>
<td>Specifies the character to be used as the escape character. Default is ~.</td>
</tr>
</tbody>
</table>
### Parameter Description

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>respect case1</td>
<td>Set to Y this will respect the case of qualifier1, qualifier2 and tablename respectively - alternative is blank.</td>
</tr>
<tr>
<td>respectcase2</td>
<td></td>
</tr>
<tr>
<td>respectcase3</td>
<td></td>
</tr>
<tr>
<td>extended_information</td>
<td>If set to Y returns additional information depending upon the data access type. Applicable to NRDB, NRDB2 and DB2 for z/OS and i5/OS only. Defaults to N (no extended information required).</td>
</tr>
<tr>
<td>access_method</td>
<td>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. 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.</td>
</tr>
<tr>
<td></td>
<td>- ADABAS - A</td>
</tr>
<tr>
<td></td>
<td>- CAPX / RT - C</td>
</tr>
<tr>
<td></td>
<td>- DATACOM - X</td>
</tr>
<tr>
<td></td>
<td>- DB2 - Z</td>
</tr>
<tr>
<td></td>
<td>- DB2390IMG - G</td>
</tr>
<tr>
<td></td>
<td>- DB2UNLD - W</td>
</tr>
<tr>
<td></td>
<td>- DL1 - D</td>
</tr>
<tr>
<td></td>
<td>- ESDS - E</td>
</tr>
<tr>
<td></td>
<td>- IDMS - I</td>
</tr>
<tr>
<td></td>
<td>- ISAM - M</td>
</tr>
<tr>
<td></td>
<td>- KSDS - K</td>
</tr>
<tr>
<td></td>
<td>- MQ - Q</td>
</tr>
<tr>
<td></td>
<td>- ODBA - O</td>
</tr>
<tr>
<td></td>
<td>- RRDS - N</td>
</tr>
<tr>
<td></td>
<td>- SEQ - S</td>
</tr>
<tr>
<td></td>
<td>- TAPE - T</td>
</tr>
<tr>
<td></td>
<td>- USER - U</td>
</tr>
</tbody>
</table>

### Related Topics:
- "DTLDESCRIBE Qualifiers" on page 268

### DTLDESCRIBE COLUMNS Returned Information

The following table describes the columns:

<table>
<thead>
<tr>
<th>Column Name</th>
<th>Description</th>
<th>Extended Information</th>
</tr>
</thead>
<tbody>
<tr>
<td>Qualifier1</td>
<td>Creator/Schema name.</td>
<td>No</td>
</tr>
<tr>
<td>Qualifier2</td>
<td>PowerExchange data map name will be returned, or null for NRDB2.</td>
<td>No</td>
</tr>
<tr>
<td>Column Name</td>
<td>Description</td>
<td>Extended Information</td>
</tr>
<tr>
<td>-------------------</td>
<td>---------------------------------------------------------------------------------------------------</td>
<td>----------------------</td>
</tr>
<tr>
<td>Tablename</td>
<td>Return table name.</td>
<td>No</td>
</tr>
<tr>
<td>Column_name</td>
<td>Column name.</td>
<td>No</td>
</tr>
<tr>
<td>Type</td>
<td>Field format.</td>
<td>No</td>
</tr>
<tr>
<td>Precision</td>
<td>Field length.</td>
<td>No</td>
</tr>
<tr>
<td>Scale</td>
<td>Decimal places.</td>
<td>No</td>
</tr>
<tr>
<td>Nullable</td>
<td>Nullable.</td>
<td>No</td>
</tr>
<tr>
<td>Detail_type</td>
<td>Internal column type representation.</td>
<td>No</td>
</tr>
<tr>
<td>Comments</td>
<td>Comments column always present. Comment information is included only if requested by specifying</td>
<td>No</td>
</tr>
<tr>
<td></td>
<td>the comments indicator in the describe.</td>
<td></td>
</tr>
<tr>
<td>Key_type</td>
<td>Reserved for future use.</td>
<td>No</td>
</tr>
<tr>
<td>Ref_qualifier_1</td>
<td>Reserved for future use.</td>
<td>No</td>
</tr>
<tr>
<td>Ref_qualifier_2</td>
<td>Reserved for future use</td>
<td>No</td>
</tr>
<tr>
<td>Ref_table_name</td>
<td>Reserved for future use</td>
<td>No</td>
</tr>
<tr>
<td>Ref_column_name</td>
<td>Reserved for future use</td>
<td>No</td>
</tr>
<tr>
<td>Base Record</td>
<td>Describes the dependencies for a table. For example, ( \text{ROOT(SEG2(SEG3,SEG4),SEG5)} ).</td>
<td>NRDB</td>
</tr>
<tr>
<td>Base Field</td>
<td>Column base field, such as \text{ROOT:ROOTKEY}.</td>
<td>NRDB</td>
</tr>
<tr>
<td>Base Field Type</td>
<td>Field type of the column's base field (CHAR, VARCHAR etc.).</td>
<td>NRDB</td>
</tr>
<tr>
<td>Base Field Offset</td>
<td>Returns the offsets of the field within the record. There is a case when no offset is returned.</td>
<td>NRDB</td>
</tr>
<tr>
<td></td>
<td>If you have an array of fields and you collapse the array when you generate the table the offset</td>
<td></td>
</tr>
<tr>
<td></td>
<td>of those fields will be 0.</td>
<td></td>
</tr>
<tr>
<td>Field Usage</td>
<td>Usage of field. For example, COMP or DISPLAY for Cobol.</td>
<td>NRDB</td>
</tr>
<tr>
<td>Field Level</td>
<td>Indent level of the field in the Cobol copybook.</td>
<td>NRDB</td>
</tr>
<tr>
<td>Original Field Name</td>
<td>Original name of the field as given in the Cobol copybook.</td>
<td>NRDB</td>
</tr>
<tr>
<td>Field Picture</td>
<td>Cobol picture format of the field.</td>
<td>NRDB</td>
</tr>
<tr>
<td>CCSID</td>
<td>CCSID.</td>
<td>DB2</td>
</tr>
<tr>
<td>Internal_{cp}</td>
<td>Internal code page.</td>
<td>DB2, NRDB, NRDB2, CAPX, and CAPXRT</td>
</tr>
</tbody>
</table>
### RELATED TOPICS:

- [“DTLDESCRIBE Qualifiers” on page 266](#)

**DTLDESCRIBE RECORDS Qualifier**

Enables you to list the available columns. Applicable to NRDB and NRDB2.

```
dtldescribe records,[qualifier1], [qualifier2], [tablename], [comments], [ORDER], [escape character], [respect case 1], [respect case 2], [respect case 3], [extended information], [access method]
```

After the basic syntax of `dtldescribe records` all the other parameters are optional. If any are coded the intervening commas must be included so that the parser can determine which field is included.

The following table describes the parameters:

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>qualifier1/2</td>
<td>Creator/schema name.</td>
</tr>
<tr>
<td>Tablename</td>
<td>This is the primary table qualifier. Records for all available tables are returned if left blank.</td>
</tr>
<tr>
<td>Comments</td>
<td>If set to Y this will return any comments data found in the database.</td>
</tr>
<tr>
<td>ORDER</td>
<td>This field must be left blank (comma needed to maintain position of later fields).</td>
</tr>
<tr>
<td>escape character</td>
<td>Specifies the character to be used as the escape character. Default is ~.</td>
</tr>
<tr>
<td>respect case 1/2/3</td>
<td>Set to Y this will respect the case of qualifier1, qualifier2 and tablename respectively - alternative is blank.</td>
</tr>
</tbody>
</table>
### Parameter Description

**extended information**  
If set to Y returns additional information depending upon the data access type.  
Applicable to NRDB, NRDB2 and DB2 for z/OS and i5/OS only.  
Defaults to N (no extended information required).

**access method**  
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.
- ADABAS - A  
- CAPX / RT - C  
- DATACOM - X  
- DB2 - Z  
- DB2390IMG - G  
- DB2UNLD - W  
- DL1 - D  
- ESDS - E  
- IDMS - I  
- ISAM - M  
- 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:

<table>
<thead>
<tr>
<th>Field</th>
<th>Returned information</th>
</tr>
</thead>
<tbody>
<tr>
<td>Record name</td>
<td>Name of the record.</td>
</tr>
</tbody>
</table>
| 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. |
<table>
<thead>
<tr>
<th>Field</th>
<th>Returned information</th>
</tr>
</thead>
<tbody>
<tr>
<td>Acc_mths03</td>
<td>Not used. Reserved for future use. The comma is needed to maintain position of subsequent fields.</td>
</tr>
<tr>
<td>Acc_mths04</td>
<td>Not used. Reserved for future use. The comma is needed to maintain position of subsequent fields.</td>
</tr>
</tbody>
</table>
| Acc_mths05  | IDMS Log-Based. The technique that IDMS uses to physically store occurrences of the record type.  
|             | - 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], [tablename], [comments], [ORDER], [escape character], [respect case 1], [respect case 2], [respect case 3], [extended information],[access method]`

After the basic syntax of dtldescribe pkeys all the other parameters are optional. If any are coded the intervening commas must be included so that the parser can determine which field is included.
The following table describes the parameters:

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>qualifier1/2</td>
<td>Returns the first and second data map qualifiers.</td>
</tr>
<tr>
<td>Tablename</td>
<td>This is the primary table qualifier. Returns the table name.</td>
</tr>
<tr>
<td>Comments</td>
<td>If set to Y this will return any comments data found in the database.</td>
</tr>
<tr>
<td>ORDER</td>
<td>This field must be left blank (comma needed to maintain position of later fields).</td>
</tr>
<tr>
<td>escape character</td>
<td>Specifies the character to be used as the escape character. Default is ~.</td>
</tr>
<tr>
<td>respect case 1/2/3</td>
<td>Set to Y this will respect the case of qualifier1, qualifier2 and tablename respectively - alternative is blank.</td>
</tr>
<tr>
<td>extended information</td>
<td>This field must be left blank (comma needed to maintain position of later fields).</td>
</tr>
<tr>
<td>access method</td>
<td>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.</td>
</tr>
</tbody>
</table>

- ADABAS - A
- CAPX / RT - C
- DATACOM - X
- DB2 - Z
- DB2390IMG - G
- DB2UNLD - W
- DL1 - D
- ESDS - E
- IDMS - I
- ISAM - M
- 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 268

**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]
```

The following table describes the parameters:

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>pk_qualifier1/2</td>
<td>Primary key qualifiers.</td>
</tr>
<tr>
<td>pk_tablename</td>
<td>This is the primary table qualifier.</td>
</tr>
<tr>
<td>fk_qualifier1/2</td>
<td>These refer to foreign key qualifiers.</td>
</tr>
<tr>
<td>fk_tablename</td>
<td>This is the foreign table qualifier.</td>
</tr>
<tr>
<td>comments</td>
<td>Set to Y to get any comments data found in the database.</td>
</tr>
<tr>
<td>ORDER</td>
<td>Must be blank (comma needed to maintain position if later arguments are used.).</td>
</tr>
<tr>
<td>escape character</td>
<td>Specifies the character to be used as the escape character. Default is ~.</td>
</tr>
<tr>
<td>respect case pk_/fk_1/2/3</td>
<td>Set to Y this will respect the case of pk_qualifier1,2 or fk_qualifier1,2 and pk_tablename or fk_tablename as appropriate - alternative is blank.</td>
</tr>
<tr>
<td>extended information</td>
<td>This field must be left blank (comma needed to maintain position of later fields).</td>
</tr>
<tr>
<td>access method</td>
<td>This field must be left blank (comma needed to maintain position of later fields).</td>
</tr>
</tbody>
</table>

FKEYS is implemented for:
• DB2
• DB2UDB
• DB2400C
• ORACLE
• DB2390IMG
• MSSQL
RELATED TOPICS:

• “DTLDESCRIBE Qualifiers” on page 268

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:

```sql
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:

```sql
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.

```sql
DTLDESCRIBE columns,SCOTT,,EMP
```

DTLDESCRIBE FKEYS Primary Keys Information Retrieval

Example:
Table MFERNANDEZ.T4 has a primary key (PK4) defined on columns T41,T42.

```sql
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

Example 1 Result:

Example 2 Result:

Example 2:

Given a "parent table", find all the foreign keys that point to that table

Example 2 Result:

It is valid to query on both the parent and the foreign tables as shown below.

DTLDESCRIBE SCHEMAS Qualifier

Enables you to list the available schemas:

After the basic syntax of dtldescribe schemas, the filter_pattern and access_method parameters are optional. All the other parameters are ignored. If any are coded the intervening commas must be included so that the parser can determine which field is included.
The following table describes the parameters:

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>filter_pattern</td>
<td>Supply a pattern to filter the schemas that are returned to just those that start with the supplied character. Use an asterisk (<em>) as a wildcard to match one or more characters of any type. For example, Dtldescribe schemas, d</em> returns: dev_schema but not eval_schema</td>
</tr>
<tr>
<td>access method</td>
<td>For NRDB, NRDB2: If no value is used, the DTLDESCRIBE defaults to return all access methods. 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, for example, OS or AKTM. If no value is used, the DTLDESCRIBE defaults to return all access method. ADABAS - A, CAPX / RT - C, DATACOM - X, DB2 - Z, DB2390IMG - G, DB2UNLD - W, DL1 - D, ESDS - E, IDMS - I, ISAM - M, KSDS - K, MQ - Q, ODBA - O, RRDS - N, SEQ - S, TAPE - T, USER - U</td>
</tr>
</tbody>
</table>

If no value is used, the DTLDESCRIBE defaults to return all access methods.

For example,

Dtldescribe schemas, d* , , , , , , R

will return:

dev_schema but not eval_schema but only if dev_schema is using the CAPXRT access method.

This function is implemented for ADABAS Unload, CAPX, CAPXRT, DB2, DB2400C, IMSUNLD, INFORMIX, 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:

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Explanation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Qualifier 1</td>
<td>Optional, Schema name</td>
</tr>
<tr>
<td>Qualifier 2</td>
<td>Optional, Table owner</td>
</tr>
</tbody>
</table>

### DTLDESCRIBE Qualifiers for DB2 for z/OS Sources

The following table lists the DTLDESCRIBE qualifiers for DB2 for z/OS sources:

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Explanation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Qualifier 1</td>
<td>Optional, Creator</td>
</tr>
<tr>
<td>Qualifier 2</td>
<td>Optional, DBName</td>
</tr>
</tbody>
</table>

### 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:

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Explanation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Qualifier 1</td>
<td>Optional, Schema</td>
</tr>
<tr>
<td>Qualifier 2</td>
<td>Optional, Definer (Creator ID)</td>
</tr>
</tbody>
</table>

### DTLDESCRIBE Qualifiers for Informix Sources

The following table lists the DTLDESCRIBE qualifiers for Informix sources:

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Explanation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Qualifier 1</td>
<td>Optional, Owner</td>
</tr>
<tr>
<td>Qualifier 2</td>
<td>Not Used</td>
</tr>
</tbody>
</table>
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:

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Explanation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Qualifier 1</td>
<td>Optional, gives the first token of the data map name</td>
</tr>
<tr>
<td>Qualifier 2</td>
<td>Optional, gives the second token of the data map name</td>
</tr>
</tbody>
</table>

DTLDESCRIBE Qualifiers for NRDB2 Sources

The following table lists the DTLDESCRIBE qualifiers for nonrelational two-tier data formats:

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Explanation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Qualifier 1</td>
<td>Optional, gives the first token of the data map name</td>
</tr>
<tr>
<td>Qualifier 2</td>
<td>Not used</td>
</tr>
</tbody>
</table>

DTLDESCRIBE Qualifiers for Oracle Sources

The following table lists the DTLDESCRIBE qualifiers for Oracle sources:

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Explanation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Qualifier 1</td>
<td>Optional, Schema name</td>
</tr>
<tr>
<td>Qualifier 2</td>
<td>Not used</td>
</tr>
</tbody>
</table>

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:

<table>
<thead>
<tr>
<th>Unsupported SQL Expression or Keyword</th>
<th>Comment</th>
</tr>
</thead>
<tbody>
<tr>
<td>CREATE</td>
<td>Creation of nonrelational tables and indexes is not supported. Define these objects by using the underlying nonrelational system.</td>
</tr>
<tr>
<td>GROUP BY</td>
<td>Aggregation is not supported.</td>
</tr>
<tr>
<td>Unsupported SQL Expression or Keyword</td>
<td>Comment</td>
</tr>
<tr>
<td>-------------------------------------</td>
<td>--------------------------------</td>
</tr>
<tr>
<td>HAVING</td>
<td>Aggregation is not supported.</td>
</tr>
<tr>
<td>JOIN</td>
<td>SQL on more than one table is not supported.</td>
</tr>
<tr>
<td>ORDER BY</td>
<td>Reordering result sets is not supported.</td>
</tr>
<tr>
<td>SELECT * FROM TABLE1 WHERE COL1 IN SELECT COL2 FROM TABLE2</td>
<td>Subselects are not supported.</td>
</tr>
<tr>
<td>SELECT COUNT(*)</td>
<td>Aggregation is not supported.</td>
</tr>
<tr>
<td>UNION</td>
<td>SQL on more than one table is not supported.</td>
</tr>
</tbody>
</table>
This chapter includes the following topics:

- **PowerExchange Globalization Overview.** 272
- **Default Code Pages Installed with PowerExchange.** 273
- **General Code Pages Available with PowerExchange.** 273
- **Source-Specific Code Pages.** 274
- **User-Defined Code Pages.** 281
- **ICUCHECK Utility.** 286
- **Code Pages Shipped with PowerExchange.** 293
- **Handling Conversion Errors and Special Situations.** 305
- **Type 1 Fallback Mappings and Multibyte Data.** 308

### 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 configuration parameter 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 to 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:

<table>
<thead>
<tr>
<th>System</th>
<th>Code Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>i5/OS</td>
<td>IBM-037</td>
</tr>
<tr>
<td>UNIX and Linux</td>
<td>ISO-8859</td>
</tr>
<tr>
<td>Windows</td>
<td>ISO-8859</td>
</tr>
<tr>
<td>z/OS</td>
<td>IBM-037</td>
</tr>
</tbody>
</table>

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 parameter 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 code 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:

<table>
<thead>
<tr>
<th>Code Page</th>
<th>Uses</th>
</tr>
</thead>
<tbody>
<tr>
<td>Control code page</td>
<td>- Names of databases, tables columns, or files.</td>
</tr>
<tr>
<td></td>
<td>- User IDs or passwords.</td>
</tr>
<tr>
<td></td>
<td>- Error messages.</td>
</tr>
<tr>
<td>Data control code page</td>
<td>- 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.</td>
</tr>
<tr>
<td></td>
<td>- Parameter data refers to data sent to the server and substituted where parameter markers (question marks) are present in Delete, Select, or Update SQL.</td>
</tr>
<tr>
<td>SQL code page</td>
<td>- Code page of SQL used to access data.</td>
</tr>
<tr>
<td></td>
<td>- Typically, the SQL code page is the same as the control code page because table names can be processed by both.</td>
</tr>
<tr>
<td></td>
<td>- The SQL code page must be sufficiently precise to handle any literals.</td>
</tr>
</tbody>
</table>

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 a 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:

<table>
<thead>
<tr>
<th>Source or Target Type</th>
<th>How Code Page Numbers Are Derived</th>
</tr>
</thead>
<tbody>
<tr>
<td>DB2 on i5/OS</td>
<td>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 parameters. Otherwise, these columns are mapped to BIN columns and can only be processed in hexadecimal format.</td>
</tr>
</tbody>
</table>
| 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.  
- For single-byte columns, the internal code page number is based on the CCSID of the column and the first `sbcs_ccsid` 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 `graphic_ccsid` 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 `mixed_ccsid` 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:  
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:  
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. |
<p>| 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. |</p>
<table>
<thead>
<tr>
<th>Source or Target Type</th>
<th>How Code Page Numbers Are Derived</th>
</tr>
</thead>
<tbody>
<tr>
<td>Oracle CDC</td>
<td>The PowerExchange internal code page for the number of columns from which changes are captured is always UTF-8.</td>
</tr>
<tr>
<td>Microsoft SQL Server</td>
<td>PowerExchange determines the internal code page number from the collation sequence of the database.</td>
</tr>
</tbody>
</table>

**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:

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>SBCS_CCSID</td>
<td>Specifies the single-byte character set.</td>
</tr>
<tr>
<td>DBCS_CCSID</td>
<td>Specifies the double-byte character set.</td>
</tr>
</tbody>
</table>

**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 DSNHDECMP load module.

**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=[Y[I]]
  [EBCDIC_CCSID=({sbcss_csid[037]
  [graphic_csid[037]
  [mixed_csid[037]])
  [ASCII_CCSID=({sbcss_csid[850]
  [graphic_csid[65534]
  [mixed_csid[65534]])
  [UNICODE_CCSID=({sbcss_csid[100]
  [graphic_csid[1208]
  [mixed_csid[1208]])
  [PLAN_CCSID={sbcss_csid
    [graphic_csid
    [mixed_csid])
  [REMAPP=(current_data_csid),(remapped_data_csid)

```

You must include the DB2 subsystem SSID and 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 84.

For each _CCSID parameter, specify a value for the following types of CCSIDs:

- A SCSCSID 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*. 

---

**Source-Specific Code Pages**

277
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,
000276 AMCCSID=942,
000277 AGCCSID=301,
000278 SCCSID=290,
000279 MCCSID=930,
000280 GCSSID=300,
000281 USCCSID=367,
000282 UMCCSID=1208,
000283 UGCCSID=1200,
000302 END
000303 /*
```

Use these CCSID values in the following PowerExchange DB2CODEPAGE parameters:

```
DB2CODEPAGE=(D91G
 ,EBCDIC_CCSID=(280,300,930)
 ,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 parameter s:

```
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 DSNHDECN 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 to direct DB2 to use the plan code pages.

- 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:

```sql
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 the were defined in the DB2 plan that was bound for PowerExchange.

In this case, specify the following parameters in the DB2CODEPAGE statement:

```sql
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:

```sql
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 `REMAP` parameter to identify the CCSID to remap and the data CCSID to use. You can define up to six `REMAP` 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:

```sql
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:

```sql
DB2CODEPAGE=(D91G,
    DB2TRANS=N,
    REMAP1=(incorrect_cp,c 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:

```sql
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 286

---

**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:

To add a user-defined ICU-compatible code page:

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 parameters, 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:

<table>
<thead>
<tr>
<th>Precision Indicator</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>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.</td>
</tr>
<tr>
<td>1</td>
<td>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.</td>
</tr>
<tr>
<td>2</td>
<td>Invalid character. The character will be replaced by the substitution character.</td>
</tr>
<tr>
<td>3</td>
<td>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.</td>
</tr>
</tbody>
</table>

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:

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](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 or bulk data movements. 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:

<table>
<thead>
<tr>
<th>System</th>
<th>Location of CNV Files</th>
</tr>
</thead>
<tbody>
<tr>
<td>i5/OS</td>
<td>UNIX style directory. The makeconv utility is run from the QSH shell.</td>
</tr>
<tr>
<td></td>
<td>Set ICUDATADIR to the directory name.</td>
</tr>
<tr>
<td>UNIX and Linux</td>
<td>CNV files are located in the PowerExchange installation directory.</td>
</tr>
<tr>
<td></td>
<td>If PowerCenter workflows are run on the machine, you must copy the CNV files to the</td>
</tr>
<tr>
<td></td>
<td>server/bin directory which also contains the RES files.</td>
</tr>
</tbody>
</table>
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.

2. 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 parameter to the DBMOVER configuration file. In this file, specify the location of the CNV file.

3. Add an ICUCONVERTER(301,ADD...) parameter to the DBMOVER configuration file. Typically, the first user-defined ICU code page uses CPN 301, the next one uses 302, and so on.

4. If the code page is used by DB2 to map a CCSID to a code page, add an ICUALIAS=(301,ADD...) parameter to the DBMOVER configuration file.

5. 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=(xxx,DELETE...) parameter to the DBMOVER configuration file. Consult the ICUCHECK reports to determine the CPN of the existing code page.

6. Run ICUCHECK again to verify that the syntax of the PowerExchange configuration parameters 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 286

**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 CPN 301, and delete existing aliases that are pointing at a different code page.

To view data from the PowerExchange Navigator:

1. In the **Database Row Test** dialog box, select **Columns in Fetch** and **Extensions**.
2. Perform a database row test.
3. Verify that the internal CPN number is 301.
4. In the **Database Row Test** dialog box, select **Data in Fetch**.
5. Run a row test of the data. The data can be accessed through Personal Metadata or an NRDB data map.
6. 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=\{USRCPPN\}
```
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:

<table>
<thead>
<tr>
<th>System</th>
<th>Location and File name</th>
</tr>
</thead>
<tbody>
<tr>
<td>i5/OS</td>
<td>USRCPnn member that is specified in the CODEPAGE parameter of the DBMOVER configuration file.</td>
</tr>
<tr>
<td>UNIX and Linux</td>
<td>USRCPnn.cp file in the directory where PowerExchange is installed.</td>
</tr>
<tr>
<td>Windows</td>
<td>USRCPnn.cp file in the directory where PowerExchange is installed.</td>
</tr>
<tr>
<td>z/OS</td>
<td>Data set that is allocated by the USRCPnn DD in the PowerExchange Listener or Client JCL.</td>
</tr>
</tbody>
</table>

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 B9 B8 B6 BE 9E 6C 6D
BC E1 E2 E3 E4 E5 E6 E7 E8 E9 E1 E2 E3 E4 E5 E6 E7
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 82 6A B5 BD B4 9A 8A BA CA AF BC
90 8F EA FA EO A0 B6 B3 9D 0A 9B 8B B7 B9 B8 AB
64 65 62 66 63 67 5A 68 74 71 72 73 78 75 76 77
7C 69 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 CO 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.
5. Select 1 in your local z/OS character set. The character 1 is x’F1’ in hexadecimal format.
6. Locate 1 in the code page ISO-8859 and find its corresponding hexadecimal value. This value is 31.
7. 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.
8. 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 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 6. PMlocale Code Pages
- Report 7. Control Counts
- Report 8. Errors

**ICUCHECK Syntax**

Use the following program to display information on the screen:

```
icheck.exe
```

On Linux, UNIX, or Windows, use the following syntax to run the ICUCHECK program and write the results to a text file:

```
icheck.exe > icheck.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

<table>
<thead>
<tr>
<th>CPN Name</th>
<th>PMlocale</th>
<th>PowerCenter name</th>
<th>Space CPN2 ICU Converter</th>
</tr>
</thead>
<tbody>
<tr>
<td>001</td>
<td>ISO-8859</td>
<td>Latin1</td>
<td>057 ISO-8859-1</td>
</tr>
<tr>
<td>002</td>
<td>IBM-1047</td>
<td>IBM1047</td>
<td>213 IBM EBCDIC US English</td>
</tr>
<tr>
<td></td>
<td>ibm-1047</td>
<td>P100-1995</td>
<td></td>
</tr>
<tr>
<td>003</td>
<td>IBM-037</td>
<td>IBM037</td>
<td>189 ibm-37_P100-1995</td>
</tr>
</tbody>
</table>

The following table describes the columns in this report:

<table>
<thead>
<tr>
<th>Column</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>CPN</td>
<td>PowerExchange code page number.</td>
</tr>
<tr>
<td>Name</td>
<td>Name of the code page. PowerExchange and PowerCenter exchange data using this code page name.</td>
</tr>
<tr>
<td>PMlocale</td>
<td>Name of the character set.</td>
</tr>
<tr>
<td>PowerCenter name</td>
<td>Code page description.</td>
</tr>
<tr>
<td>Space</td>
<td>Hexadecimal value of the space character used to pad CHAR columns.</td>
</tr>
</tbody>
</table>
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
A1   : 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 parseable 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 A1 L Space Flags PMlocale PowerCenter Name
```
The following table describes the fields in the report:

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>CPN</td>
<td>PowerExchange code page number.</td>
</tr>
<tr>
<td>Converter name</td>
<td>ICU converter name.</td>
</tr>
<tr>
<td>Min</td>
<td>Minimum bytes per character.</td>
</tr>
<tr>
<td>Max</td>
<td>Maximum bytes per character.</td>
</tr>
<tr>
<td>In</td>
<td>Increment added to the buffer size in addition to the standard formula. This increment allows space for leading escape sequences on code pages.</td>
</tr>
<tr>
<td>Al</td>
<td>Number of aliases for this code page.</td>
</tr>
<tr>
<td>L</td>
<td>Length of the space character.</td>
</tr>
<tr>
<td>Space</td>
<td>Hexadecimal value of the space character that is used to pad CHAR columns.</td>
</tr>
<tr>
<td>Flag 1 : States</td>
<td>Set to S if state bytes are used.</td>
</tr>
<tr>
<td>Flag 2 : Endianness unknown</td>
<td>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.</td>
</tr>
<tr>
<td>Flag 3 : ASCII / EBDIC</td>
<td>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.</td>
</tr>
<tr>
<td>Flag 4 : Parsable without translation</td>
<td>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.</td>
</tr>
<tr>
<td>Flag 5 : Lower case non-standard</td>
<td>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.</td>
</tr>
<tr>
<td>Flag 6 : Supplemental Characters</td>
<td>Set to X if the code page includes supplemental characters beyond UCS_2.</td>
</tr>
</tbody>
</table>
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 supplemnetals
042 UTF-16 UTF16 contains supplemnetals
043 UTF-16BE UTF16_BigEnd. contains supplemnetals
```

The following table describes the fields in the report:

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>CPN</td>
<td>PowerExchange code page number.</td>
</tr>
<tr>
<td>Converter name</td>
<td>ICU converter name.</td>
</tr>
<tr>
<td>Convert. type</td>
<td>Type of the ICU converter.</td>
</tr>
<tr>
<td>Min</td>
<td>Minimum bytes for each character according to the ICU API, if different from the PowerExchange value.</td>
</tr>
</tbody>
</table>

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

<table>
<thead>
<tr>
<th>Converter name</th>
<th>Min</th>
<th>Max</th>
<th>Converter type</th>
<th>CPN</th>
</tr>
</thead>
<tbody>
<tr>
<td>UTF-8</td>
<td>1</td>
<td>3</td>
<td>UTF8</td>
<td>041</td>
</tr>
<tr>
<td>UTF-16</td>
<td>2</td>
<td>2</td>
<td>UTF16</td>
<td>042</td>
</tr>
<tr>
<td>UTF-16BE</td>
<td>2</td>
<td>2</td>
<td>UTF16_BigEndian</td>
<td>043</td>
</tr>
</tbody>
</table>

The following table describes the columns in the report:

<table>
<thead>
<tr>
<th>Column</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Converter name</td>
<td>ICU converter name.</td>
</tr>
<tr>
<td>Min</td>
<td>Minimum bytes for each character according to the ICU API, if different from the PowerExchange value.</td>
</tr>
<tr>
<td>Max</td>
<td>Maximum bytes for each character according to the ICU API, if different from the PowerExchange value.</td>
</tr>
<tr>
<td>Converter type</td>
<td>Type of ICU converter.</td>
</tr>
<tr>
<td>CPN</td>
<td>PowerExchange code page number.</td>
</tr>
</tbody>
</table>


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:
  
  \[ \text{CODEPAGE}=(\text{IBM-37,CP930,CP930}) \]

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 ICU ALIAS parameter to change the alias definitions.

The following sample report shows PowerExchange code page names and aliases.

<table>
<thead>
<tr>
<th>Title</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>CPN</td>
<td>PowerExchange code page number.</td>
</tr>
<tr>
<td>Alias name</td>
<td>ICU converter name.</td>
</tr>
<tr>
<td>Alias type</td>
<td>Type of code page alias.</td>
</tr>
</tbody>
</table>

The following table describes the columns in the report:

<table>
<thead>
<tr>
<th>Title</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>id name</td>
<td>description</td>
</tr>
<tr>
<td>1 US-ASCII</td>
<td>7-bit ASCII</td>
</tr>
<tr>
<td>4 Latin1</td>
<td>ISO 8859-1 Western European</td>
</tr>
<tr>
<td>5 ISO-8859-2</td>
<td>ISO 8859-2 Eastern European</td>
</tr>
<tr>
<td>6 ISO-8859-3</td>
<td>ISO 8859-3 Southeast European</td>
</tr>
</tbody>
</table>

Report 6. PM Locale Code Pages

The following sample report shows PM locale code pages.

<table>
<thead>
<tr>
<th>Title</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>id name</td>
<td>description</td>
</tr>
<tr>
<td>1 US-ASCII</td>
<td>7-bit ASCII</td>
</tr>
<tr>
<td>4 Latin1</td>
<td>ISO 8859-1 Western European</td>
</tr>
<tr>
<td>5 ISO-8859-2</td>
<td>ISO 8859-2 Eastern European</td>
</tr>
<tr>
<td>6 ISO-8859-3</td>
<td>ISO 8859-3 Southeast European</td>
</tr>
</tbody>
</table>

...
The following table describes the columns in the report:

<table>
<thead>
<tr>
<th>Column</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>id</td>
<td>ID number of the code page.</td>
</tr>
<tr>
<td>name</td>
<td>Name of the code page.</td>
</tr>
<tr>
<td>description</td>
<td>Description of the code page.</td>
</tr>
<tr>
<td>PWX codepages</td>
<td>Equivalent PowerExchange code page.</td>
</tr>
</tbody>
</table>

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

<table>
<thead>
<tr>
<th>Code Page Description</th>
<th>Count</th>
</tr>
</thead>
<tbody>
<tr>
<td>CPN 001-030: Simple code pages</td>
<td>16</td>
</tr>
<tr>
<td>CPN 031-040: User-defined Simple code pages</td>
<td>10</td>
</tr>
<tr>
<td>CPN 041-300: ICU code pages</td>
<td>235</td>
</tr>
<tr>
<td>CPN 301-340: User-defined ICU code pages</td>
<td>0</td>
</tr>
<tr>
<td>CPN 001-340: Total code pages</td>
<td>261</td>
</tr>
<tr>
<td>Simple code pages with PMlocales</td>
<td>16</td>
</tr>
<tr>
<td>ICU code pages with PMlocales</td>
<td>212</td>
</tr>
<tr>
<td>Total code pages with PMLocales</td>
<td>228</td>
</tr>
<tr>
<td>Most aliases for a code page</td>
<td>11</td>
</tr>
<tr>
<td>Code page with the most aliases</td>
<td>43 (UTF-16BE)</td>
</tr>
</tbody>
</table>

PowerCenter PMLocales not mapped to PWX : 32

Report 8. Errors

The following report shows errors and warnings.

Report 8: Errors

<table>
<thead>
<tr>
<th>Description</th>
<th>Count</th>
</tr>
</thead>
<tbody>
<tr>
<td>PWX PMLocales not in PowerCenter</td>
<td>3</td>
</tr>
<tr>
<td>Total errors</td>
<td>3</td>
</tr>
</tbody>
</table>

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/usersguide/icudata.html.

PowerExchange Static Code Pages

The following table lists the single-byte static code pages that are shipped with PowerExchange:

<table>
<thead>
<tr>
<th>Code Page</th>
<th>Typical Use</th>
</tr>
</thead>
<tbody>
<tr>
<td>ISO-8859</td>
<td>UNIX and Windows</td>
</tr>
<tr>
<td>IBM-037</td>
<td>USA, Canada, Brazil</td>
</tr>
<tr>
<td>IBM-273</td>
<td>Germany, Austria</td>
</tr>
<tr>
<td>IBM-277</td>
<td>Denmark, Norway</td>
</tr>
<tr>
<td>IBM-278</td>
<td>Finland, Sweden</td>
</tr>
<tr>
<td>IBM-280</td>
<td>Italy</td>
</tr>
<tr>
<td>IBM-284</td>
<td>Spain, Latin America</td>
</tr>
<tr>
<td>IBM-285</td>
<td>United Kingdom</td>
</tr>
<tr>
<td>IBM-297</td>
<td>France</td>
</tr>
<tr>
<td>IBM-424</td>
<td>Modern Hebrew</td>
</tr>
<tr>
<td>IBM-500</td>
<td>International</td>
</tr>
<tr>
<td>IBM-870</td>
<td>Polish</td>
</tr>
<tr>
<td>IBM-1047</td>
<td>Latin 1/Open Systems</td>
</tr>
<tr>
<td>MS-1250</td>
<td>PC Regional options setting for Central Europe (Polish)</td>
</tr>
</tbody>
</table>

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:

<table>
<thead>
<tr>
<th>Name</th>
<th>Typical Use</th>
<th>Character Length in Bytes</th>
</tr>
</thead>
<tbody>
<tr>
<td>BOCU-1</td>
<td>Email</td>
<td>3+</td>
</tr>
<tr>
<td>CESU-8</td>
<td>Compression</td>
<td>3+</td>
</tr>
<tr>
<td>IMAP-mailbox-name</td>
<td>Email</td>
<td>3+</td>
</tr>
<tr>
<td>Name</td>
<td>Typical Use</td>
<td>Character Length in Bytes</td>
</tr>
<tr>
<td>---------------------</td>
<td>-------------</td>
<td>---------------------------</td>
</tr>
<tr>
<td>SCSU</td>
<td>Compression</td>
<td>3+</td>
</tr>
<tr>
<td>UTF-16</td>
<td>-</td>
<td>2</td>
</tr>
<tr>
<td>UTF16_OppositeEndian</td>
<td>-</td>
<td>2</td>
</tr>
<tr>
<td>UTF16_PlatformEndian</td>
<td>-</td>
<td>2</td>
</tr>
<tr>
<td>UTF-16BE</td>
<td>Databases</td>
<td>2</td>
</tr>
<tr>
<td>UTF-16LE</td>
<td>Databases</td>
<td>2</td>
</tr>
<tr>
<td>UTF-32</td>
<td>-</td>
<td>4</td>
</tr>
<tr>
<td>UTF32_OppositeEndian</td>
<td>-</td>
<td>4</td>
</tr>
<tr>
<td>UTF32_PlatformEndian</td>
<td>-</td>
<td>4</td>
</tr>
<tr>
<td>UTF-32BE</td>
<td>-</td>
<td>4</td>
</tr>
<tr>
<td>UTF-32LE</td>
<td>-</td>
<td>4</td>
</tr>
<tr>
<td>UTF-7</td>
<td>Email</td>
<td>3+</td>
</tr>
<tr>
<td>UTF-8</td>
<td>Databases</td>
<td>3+</td>
</tr>
</tbody>
</table>

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':

<table>
<thead>
<tr>
<th>Name</th>
<th>Typical Use</th>
<th>Character Length ins</th>
</tr>
</thead>
<tbody>
<tr>
<td>ibm-1089_P100-1995</td>
<td>Arabic</td>
<td>1</td>
</tr>
<tr>
<td>ibm-1256_P110-1997</td>
<td>Arabic</td>
<td>1</td>
</tr>
<tr>
<td>ibm-5352_P100-1998</td>
<td>Arabic</td>
<td>1</td>
</tr>
<tr>
<td>windows-1256-2000</td>
<td>Arabic</td>
<td>1</td>
</tr>
<tr>
<td>Name</td>
<td>Typical Use</td>
<td>Character Length ins</td>
</tr>
<tr>
<td>--------------------</td>
<td>---------------------</td>
<td>----------------------</td>
</tr>
<tr>
<td>ibm-1257_P100-1995</td>
<td>Baltic</td>
<td>1</td>
</tr>
<tr>
<td>ibm-5353_P100-1998</td>
<td>Baltic</td>
<td>1</td>
</tr>
<tr>
<td>ibm-901_P100-1999</td>
<td>Baltic</td>
<td>1</td>
</tr>
<tr>
<td>ibm-914_P100-1995</td>
<td>Baltic</td>
<td>1</td>
</tr>
<tr>
<td>ibm-921_P100-1995</td>
<td>Baltic</td>
<td>1</td>
</tr>
<tr>
<td>ibm-9449_P100-2002</td>
<td>Baltic</td>
<td>1</td>
</tr>
<tr>
<td>ibm-1131_P100-1997</td>
<td>Belarusian</td>
<td>1</td>
</tr>
<tr>
<td>ibm-1250_P100-1995</td>
<td>Central &amp; East Europe</td>
<td>1</td>
</tr>
<tr>
<td>ibm-912_P100-1995</td>
<td>Central &amp; East Europe</td>
<td>1</td>
</tr>
<tr>
<td>ibm-913_P100-2000</td>
<td>Central &amp; East Europe</td>
<td>1</td>
</tr>
<tr>
<td>ibm-5346_P100-1998</td>
<td>Central &amp; East Europe</td>
<td>1</td>
</tr>
<tr>
<td>gb18030</td>
<td>Chinese</td>
<td>3+</td>
</tr>
<tr>
<td>ibm-964_P110-1999</td>
<td>Chinese</td>
<td>3+</td>
</tr>
<tr>
<td>HZ</td>
<td>Chinese (simplified)</td>
<td>3+</td>
</tr>
<tr>
<td>ibm-1381_P110-1999</td>
<td>Chinese (simplified)</td>
<td>1 to 2</td>
</tr>
<tr>
<td>ibm-1383_P110-1999</td>
<td>Chinese (simplified)</td>
<td>1 to 2</td>
</tr>
<tr>
<td>ibm-1386_P100-2002</td>
<td>Chinese (simplified)</td>
<td>1 to 2</td>
</tr>
<tr>
<td>ibm-5478_P100-1995</td>
<td>Chinese (simplified)</td>
<td>2</td>
</tr>
<tr>
<td>windows-936-2000</td>
<td>Chinese (simplified)</td>
<td>1 to 2</td>
</tr>
<tr>
<td>ibm-1373_P100-2002</td>
<td>Chinese (traditional)</td>
<td>1 to 2</td>
</tr>
<tr>
<td>ibm-1375_P100-2003</td>
<td>Chinese (traditional)</td>
<td>1 to 2</td>
</tr>
<tr>
<td>ibm-950_P110-1999</td>
<td>Chinese (traditional)</td>
<td>1 to 2</td>
</tr>
<tr>
<td>windows-950-2000</td>
<td>Chinese (traditional)</td>
<td>1 to 2</td>
</tr>
<tr>
<td>ibm-1251_P100-1995</td>
<td>Cyrillic</td>
<td>1</td>
</tr>
<tr>
<td>ibm-5347_P100-1998</td>
<td>Cyrillic</td>
<td>1</td>
</tr>
<tr>
<td>ibm-915_P100-1995</td>
<td>Cyrillic</td>
<td>1</td>
</tr>
<tr>
<td>ibm-902_P100-1999</td>
<td>Estonian</td>
<td>1</td>
</tr>
<tr>
<td>Name</td>
<td>Typical Use</td>
<td>Character Length ins</td>
</tr>
<tr>
<td>-----------------------</td>
<td>-------------</td>
<td>----------------------</td>
</tr>
<tr>
<td>ibm-922_P100-1999</td>
<td>Estonian</td>
<td>1</td>
</tr>
<tr>
<td>ibm-923_P100-1998</td>
<td>Estonian</td>
<td>1</td>
</tr>
<tr>
<td>ibm-1098_P100-1995</td>
<td>Farsi</td>
<td>1</td>
</tr>
<tr>
<td>ibm-1253_P100-1995</td>
<td>Greek</td>
<td>1</td>
</tr>
<tr>
<td>ibm-4909_P100-1999</td>
<td>Greek</td>
<td>1</td>
</tr>
<tr>
<td>ibm-5349_P100-1998</td>
<td>Greek</td>
<td>1</td>
</tr>
<tr>
<td>ibm-813_P100-1995</td>
<td>Greek</td>
<td>1</td>
</tr>
<tr>
<td>ibm-1255_P100-1995</td>
<td>Hebrew</td>
<td>1</td>
</tr>
<tr>
<td>ibm-5351_P100-1998</td>
<td>Hebrew</td>
<td>1</td>
</tr>
<tr>
<td>ibm-916_P100-1995</td>
<td>Hebrew</td>
<td>1</td>
</tr>
<tr>
<td>ibm-9447_P100-2002</td>
<td>Hebrew</td>
<td>1</td>
</tr>
<tr>
<td>ibm-1006_P100-1995</td>
<td>India</td>
<td>1</td>
</tr>
<tr>
<td>ibm-33722_P120-1999</td>
<td>Japanese</td>
<td>3+</td>
</tr>
<tr>
<td>ibm-33722_P12A-1999</td>
<td>Japanese</td>
<td>3+</td>
</tr>
<tr>
<td>ibm-942_P12A-1999</td>
<td>Japanese</td>
<td>1 to 2</td>
</tr>
<tr>
<td>ibm-943_P130-1999</td>
<td>Japanese</td>
<td>1 to 2</td>
</tr>
<tr>
<td>ibm-943_P15A-2003</td>
<td>Japanese</td>
<td>1 to 2</td>
</tr>
<tr>
<td>ibm-954_P101-2000</td>
<td>Japanese</td>
<td>3+</td>
</tr>
<tr>
<td>ibm-1363_P110-1997</td>
<td>Korean</td>
<td>1 to 2</td>
</tr>
<tr>
<td>ibm-1363_P11B-1998</td>
<td>Korean</td>
<td>1 to 2</td>
</tr>
<tr>
<td>ibm-949_P110-1999</td>
<td>Korean</td>
<td>1 to 2</td>
</tr>
<tr>
<td>ibm-949_P11A-1999</td>
<td>Korean</td>
<td>1 to 2</td>
</tr>
<tr>
<td>ibm-970_P110-1995</td>
<td>Korean</td>
<td>1 to 2</td>
</tr>
<tr>
<td>ibm-971_P100-1995</td>
<td>Korean</td>
<td>2</td>
</tr>
<tr>
<td>windows-1361-2000</td>
<td>Korean</td>
<td>1 to 2</td>
</tr>
<tr>
<td>windows-949-2000</td>
<td>Korean</td>
<td>1 to 2</td>
</tr>
<tr>
<td>ibm-1133_P100-1997</td>
<td>Lao</td>
<td>1</td>
</tr>
<tr>
<td>Name</td>
<td>Typical Use</td>
<td>Character Length ins</td>
</tr>
<tr>
<td>-----------------------</td>
<td>----------------------</td>
<td>----------------------</td>
</tr>
<tr>
<td>iso-8859_10-1998</td>
<td>Swedish</td>
<td>1</td>
</tr>
<tr>
<td>ibm-1162_P100-1999</td>
<td>Thai</td>
<td>1</td>
</tr>
<tr>
<td>ibm-874_P100-1995</td>
<td>Thai</td>
<td>1</td>
</tr>
<tr>
<td>windows-874-2000</td>
<td>Thai</td>
<td>1</td>
</tr>
<tr>
<td>ibm-1254_P100-1995</td>
<td>Turkish</td>
<td>1</td>
</tr>
<tr>
<td>ibm-5350_P100-1998</td>
<td>Turkish</td>
<td>1</td>
</tr>
<tr>
<td>ibm-920_P100-1995</td>
<td>Turkish</td>
<td>1</td>
</tr>
<tr>
<td>ibm-1124_P100-1996</td>
<td>Ukrainian</td>
<td>1</td>
</tr>
<tr>
<td>ibm-1125_P100-1997</td>
<td>Ukrainian</td>
<td>1</td>
</tr>
<tr>
<td>ibm-1051_P100-1995</td>
<td>US and international</td>
<td>1</td>
</tr>
<tr>
<td>ibm-1252_P100-2000</td>
<td>US and international</td>
<td>1</td>
</tr>
<tr>
<td>ibm-367_P100-1995</td>
<td>US and international</td>
<td>1</td>
</tr>
<tr>
<td>ibm-5348_P100-1997</td>
<td>US and international</td>
<td>1</td>
</tr>
<tr>
<td>ISO-8859-1</td>
<td>US and international</td>
<td>1</td>
</tr>
<tr>
<td>US-ASCII</td>
<td>US and international</td>
<td>1</td>
</tr>
<tr>
<td>ibm-1129_P100-1997</td>
<td>Vietnamese</td>
<td>1</td>
</tr>
<tr>
<td>ibm-1258_P100-1997</td>
<td>Vietnamese</td>
<td>1</td>
</tr>
<tr>
<td>ibm-5354_P100-1998</td>
<td>Vietnamese</td>
<td>1</td>
</tr>
</tbody>
</table>

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:

<table>
<thead>
<tr>
<th>Name</th>
<th>Typical Use</th>
<th>Character Length in Bytes</th>
</tr>
</thead>
<tbody>
<tr>
<td>ibm-16804_X110-1999</td>
<td>Arabic</td>
<td>1</td>
</tr>
<tr>
<td>ibm-16804_X110-1999,swaplfnl</td>
<td>Arabic</td>
<td>1</td>
</tr>
<tr>
<td>ibm-420_X120-1999</td>
<td>Arabic</td>
<td>1</td>
</tr>
<tr>
<td>ibm-918_P100-1995</td>
<td>Arabic</td>
<td>1</td>
</tr>
<tr>
<td>ibm-1112_P100-1995</td>
<td>Baltic</td>
<td>1</td>
</tr>
<tr>
<td>ibm-500_P100-1995</td>
<td>Belgium, Switzerland</td>
<td>1</td>
</tr>
<tr>
<td>ibm-1156_P100-1999</td>
<td>Central &amp; East Europe</td>
<td>1</td>
</tr>
<tr>
<td>ibm-13124_P100-1995</td>
<td>Chinese (simplified)</td>
<td>1</td>
</tr>
<tr>
<td>ibm-1388_P103-2001</td>
<td>Chinese (simplified)</td>
<td>1 to 2</td>
</tr>
<tr>
<td>ibm-4933_P100-2002</td>
<td>Chinese (simplified)</td>
<td>2</td>
</tr>
<tr>
<td>ibm-836_P100-1995</td>
<td>Chinese (simplified)</td>
<td>1</td>
</tr>
<tr>
<td>ibm-837_X100-1995</td>
<td>Chinese (simplified)</td>
<td>2</td>
</tr>
<tr>
<td>ibm-935_P110-1999</td>
<td>Chinese (simplified)</td>
<td>1 to 2</td>
</tr>
<tr>
<td>ibm-1159_P100-1999</td>
<td>Chinese (traditional)</td>
<td>1</td>
</tr>
<tr>
<td>ibm-1371_P100-1999</td>
<td>Chinese (traditional)</td>
<td>1 to 2</td>
</tr>
<tr>
<td>ibm-835_X100-1995</td>
<td>Chinese (traditional)</td>
<td>2</td>
</tr>
<tr>
<td>ibm-937_P110-1999</td>
<td>Chinese (traditional)</td>
<td>1 to 2</td>
</tr>
<tr>
<td>ibm-1025_P100-1995</td>
<td>Cyrillic</td>
<td>1</td>
</tr>
<tr>
<td>ibm-1123_P100-1995</td>
<td>Cyrillic</td>
<td>1</td>
</tr>
<tr>
<td>ibm-1154_P100-1999</td>
<td>Cyrillic</td>
<td>1</td>
</tr>
<tr>
<td>ibm-1158_P100-1999</td>
<td>Cyrillic</td>
<td>1</td>
</tr>
<tr>
<td>ibm-1142_P100-1997</td>
<td>Danish</td>
<td>1</td>
</tr>
<tr>
<td>ibm-1142_P100-1997,swaplfnl</td>
<td>Danish</td>
<td>1</td>
</tr>
<tr>
<td>ibm-277_P100-1995</td>
<td>Danish</td>
<td>1</td>
</tr>
<tr>
<td>ibm-1137_P100-1999</td>
<td>Devanagari</td>
<td>1</td>
</tr>
<tr>
<td>ibm-1122_P100-1999</td>
<td>Estonian</td>
<td>1</td>
</tr>
<tr>
<td>ibm-1157_P100-1999</td>
<td>Estonian</td>
<td>1</td>
</tr>
<tr>
<td>Name</td>
<td>Typical Use</td>
<td>Character Length in Bytes</td>
</tr>
<tr>
<td>-----------------------------</td>
<td>-------------</td>
<td>---------------------------</td>
</tr>
<tr>
<td>ibm-1097_P100-1995</td>
<td>Farsi</td>
<td>1</td>
</tr>
<tr>
<td>ibm-1147_P100-1997</td>
<td>French</td>
<td>1</td>
</tr>
<tr>
<td>ibm-1147_P100-1997,swapfnl</td>
<td>French</td>
<td>1</td>
</tr>
<tr>
<td>ibm-297_P100-1995</td>
<td>French</td>
<td>1</td>
</tr>
<tr>
<td>ibm-1141_P100-1997</td>
<td>German</td>
<td>1</td>
</tr>
<tr>
<td>ibm-273_P100-1995</td>
<td>German</td>
<td>1</td>
</tr>
<tr>
<td>ibm-4971_P100-1999</td>
<td>Greek</td>
<td>1</td>
</tr>
<tr>
<td>ibm-875_P100-1995</td>
<td>Greek</td>
<td>1</td>
</tr>
<tr>
<td>ibm-12712_P100-1998</td>
<td>Hebrew</td>
<td>1</td>
</tr>
<tr>
<td>ibm-12712_P100-1998,swapfnl</td>
<td>Hebrew</td>
<td>1</td>
</tr>
<tr>
<td>ibm-424_P100-1995</td>
<td>Hebrew</td>
<td>1</td>
</tr>
<tr>
<td>ibm-4899_P100-1998</td>
<td>Hebrew</td>
<td>1</td>
</tr>
<tr>
<td>ibm-803_P100-1999</td>
<td>Hebrew</td>
<td>1</td>
</tr>
<tr>
<td>ibm-1149_P100-1997</td>
<td>Icelandic</td>
<td>1</td>
</tr>
<tr>
<td>ibm-1149_P100-1997,swapfnl</td>
<td>Icelandic</td>
<td>1</td>
</tr>
<tr>
<td>ibm-871_P100-1995</td>
<td>Icelandic</td>
<td>1</td>
</tr>
<tr>
<td>ibm-1144_P100-1997</td>
<td>Italian</td>
<td>1</td>
</tr>
<tr>
<td>ibm-1144_P100-1997,swapfnl</td>
<td>Italian</td>
<td>1</td>
</tr>
<tr>
<td>ibm-280_P100-1995</td>
<td>Italian</td>
<td>1</td>
</tr>
<tr>
<td>ibm-1390_P110-2003</td>
<td>Japanese</td>
<td>1 to 2</td>
</tr>
<tr>
<td>ibm-1399_P110-2003</td>
<td>Japanese</td>
<td>1 to 2</td>
</tr>
<tr>
<td>ibm-16684_P110-2003</td>
<td>Japanese</td>
<td>2</td>
</tr>
<tr>
<td>ibm-290_P100-1995</td>
<td>Japanese</td>
<td>1</td>
</tr>
<tr>
<td>ibm-5123_P100-1999</td>
<td>Japanese</td>
<td>1</td>
</tr>
<tr>
<td>ibm-8482_P100-1999</td>
<td>Japanese</td>
<td>1</td>
</tr>
<tr>
<td>ibm-930_P120-1999</td>
<td>Japanese</td>
<td>1 to 2</td>
</tr>
<tr>
<td>ibm-939_P120-1999</td>
<td>Japanese</td>
<td>1 to 2</td>
</tr>
<tr>
<td>Name</td>
<td>Typical Use</td>
<td>Character Length in Bytes</td>
</tr>
<tr>
<td>---------------------</td>
<td>-------------</td>
<td>---------------------------</td>
</tr>
<tr>
<td>infa-FujitsuJEF</td>
<td>Japanese</td>
<td>1 to 2</td>
</tr>
<tr>
<td>infa-FujitsuJEF_kana</td>
<td>Japanese</td>
<td>1 to 2</td>
</tr>
<tr>
<td>infa-HitachiKEIS</td>
<td>Japanese</td>
<td>1 to 2</td>
</tr>
<tr>
<td>infa-HitachiKEIS_kana</td>
<td>Japanese</td>
<td>1 to 2</td>
</tr>
<tr>
<td>infa-jipse</td>
<td>Japanese</td>
<td>1 to 2</td>
</tr>
<tr>
<td>infa-jipse_kana</td>
<td>Japanese</td>
<td>1 to 2</td>
</tr>
<tr>
<td>infa-jp_ebcdic</td>
<td>Japanese</td>
<td>1 to 2</td>
</tr>
<tr>
<td>infa-jp_ebcdik</td>
<td>Japanese</td>
<td>1 to 2</td>
</tr>
<tr>
<td>infa-melcom</td>
<td>Japanese</td>
<td>1 to 2</td>
</tr>
<tr>
<td>infa-melcom_kana</td>
<td>Japanese</td>
<td>1 to 2</td>
</tr>
<tr>
<td>infa-unisys</td>
<td>Japanese</td>
<td>1 to 2</td>
</tr>
<tr>
<td>infa-unisys_kana</td>
<td>Japanese</td>
<td>1 to 2</td>
</tr>
<tr>
<td>ibm-13121_P100-1995</td>
<td>Korean</td>
<td>1</td>
</tr>
<tr>
<td>ibm-1364_P110-1997</td>
<td>Korean</td>
<td>1 to 2</td>
</tr>
<tr>
<td>ibm-833_P100-1995</td>
<td>Korean</td>
<td>1</td>
</tr>
<tr>
<td>ibm-834_P100-1995</td>
<td>Korean</td>
<td>2</td>
</tr>
<tr>
<td>ibm-933_P110-1995</td>
<td>Korean</td>
<td>1 to 2</td>
</tr>
<tr>
<td>ibm-1132_P100-1998</td>
<td>Lao</td>
<td>1</td>
</tr>
<tr>
<td>ibm-870_P100-1995</td>
<td>Polish</td>
<td>1</td>
</tr>
<tr>
<td>ibm-1153_P100-1999</td>
<td>Romanian</td>
<td>1</td>
</tr>
<tr>
<td>ibm-1153_P100-1999,swaplfnl</td>
<td>Romanian</td>
<td>1</td>
</tr>
<tr>
<td>ibm-1145_P100-1997</td>
<td>Spanish</td>
<td>1</td>
</tr>
<tr>
<td>ibm-1145_P100-1997,swaplfnl</td>
<td>Spanish</td>
<td>1</td>
</tr>
<tr>
<td>ibm-284_P100-1995</td>
<td>Spanish</td>
<td>1</td>
</tr>
<tr>
<td>ibm-1143_P100-1997</td>
<td>Swedish</td>
<td>1</td>
</tr>
<tr>
<td>ibm-1143_P100-1997,swaplfnl</td>
<td>Swedish</td>
<td>1</td>
</tr>
<tr>
<td>ibm-278_P100-1995</td>
<td>Swedish</td>
<td>1</td>
</tr>
<tr>
<td>Name</td>
<td>Typical Use</td>
<td>Character Length in Bytes</td>
</tr>
<tr>
<td>------------------------------------</td>
<td>-------------</td>
<td>----------------------------</td>
</tr>
<tr>
<td>ibm-1160_P100-1999</td>
<td>Thai</td>
<td>1</td>
</tr>
<tr>
<td>ibm-838_P100-1995</td>
<td>Thai</td>
<td>1</td>
</tr>
<tr>
<td>ibm-1026_P100-1995</td>
<td>Turkish</td>
<td>1</td>
</tr>
<tr>
<td>ibm-1155_P100-1999</td>
<td>Turkish</td>
<td>1</td>
</tr>
<tr>
<td>ibm-1146_P100-1997</td>
<td>UK English</td>
<td>1</td>
</tr>
<tr>
<td>ibm-1146_P100-1997.swaplfnl</td>
<td>UK English</td>
<td>1</td>
</tr>
<tr>
<td>ibm-285_P100-1995</td>
<td>UK English</td>
<td>1</td>
</tr>
<tr>
<td>ebcdic-xml-us</td>
<td>US and international</td>
<td>1</td>
</tr>
<tr>
<td>ibm-1047_P100-1995</td>
<td>US and international</td>
<td>1</td>
</tr>
<tr>
<td>ibm-1047_P100-1995.swaplfnl</td>
<td>US and international</td>
<td>1</td>
</tr>
<tr>
<td>ibm-1140_P100-1997</td>
<td>US and international</td>
<td>1</td>
</tr>
<tr>
<td>ibm-1140_P100-1997.swaplfnl</td>
<td>US and international</td>
<td>1</td>
</tr>
<tr>
<td>ibm-1148_P100-1997</td>
<td>US and international</td>
<td>1</td>
</tr>
<tr>
<td>ibm-1148_P100-1997.swaplfnl</td>
<td>US and international</td>
<td>1</td>
</tr>
<tr>
<td>ibm-37_P100-1995</td>
<td>US and international</td>
<td>1</td>
</tr>
<tr>
<td>ibm-37_P100-1995.swaplfnl</td>
<td>US and international</td>
<td>1</td>
</tr>
<tr>
<td>ibm-1130_P100-1997</td>
<td>Vietnamese</td>
<td>1</td>
</tr>
<tr>
<td>ibm-1164_P100-1999</td>
<td>Vietnamese</td>
<td>1</td>
</tr>
</tbody>
</table>

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 \((16684 = 300 + 4096 \times 4)\)
The following table lists CCSIDs that you can use in the same DB2 table on i5/OS or z/OS:

<table>
<thead>
<tr>
<th>Typical Use</th>
<th>Single-Byte SCCSID Columns</th>
<th>Double-Byte GCCSID Columns</th>
<th>Mixed-Length MCCSID Columns</th>
</tr>
</thead>
<tbody>
<tr>
<td>Japan - Extended English</td>
<td>1027</td>
<td>300</td>
<td>939</td>
</tr>
<tr>
<td>Japan - Extended Katakana</td>
<td>290</td>
<td>300</td>
<td>930</td>
</tr>
<tr>
<td>Japan - Katakana - Kanji</td>
<td>8482</td>
<td>16684</td>
<td>1390</td>
</tr>
<tr>
<td>Japan - Latin - Kanji</td>
<td>5123</td>
<td>16684</td>
<td>1399</td>
</tr>
<tr>
<td>Korean</td>
<td>833</td>
<td>834</td>
<td>933</td>
</tr>
<tr>
<td>Korean</td>
<td>13121</td>
<td>4390</td>
<td>1364</td>
</tr>
<tr>
<td>Simplified Chinese</td>
<td>836</td>
<td>837</td>
<td>935</td>
</tr>
<tr>
<td>Simplified Chinese</td>
<td>13124</td>
<td>4933</td>
<td>1388</td>
</tr>
<tr>
<td>Traditional Chinese</td>
<td>28709</td>
<td>835</td>
<td>937</td>
</tr>
<tr>
<td>Traditional Chinese (IBM Big-5)</td>
<td>1114</td>
<td>947</td>
<td>950</td>
</tr>
</tbody>
</table>

**Specialized ICU Code Pages**

The following table lists ICU code pages that are used in specialized contexts:

<table>
<thead>
<tr>
<th>Name</th>
<th>Typical Use</th>
<th>Character Length in Bytes</th>
</tr>
</thead>
<tbody>
<tr>
<td>ibm-1276_P100-1995</td>
<td>Adobe</td>
<td>1</td>
</tr>
<tr>
<td>ibm-1277_P100-1995</td>
<td>Adobe</td>
<td>1</td>
</tr>
<tr>
<td>macos-0_2-10.2</td>
<td>Apple - US &amp; International</td>
<td>1</td>
</tr>
<tr>
<td>macos-2566-10.2</td>
<td>Apple - East Asia</td>
<td>1 to 2</td>
</tr>
<tr>
<td>macos-29-10.2</td>
<td>Apple - Central &amp; East Europe</td>
<td>1</td>
</tr>
<tr>
<td>macos-35-10.2</td>
<td>Apple -Turkish</td>
<td>1</td>
</tr>
<tr>
<td>macos-6-10.2</td>
<td>Apple - Greek</td>
<td>1</td>
</tr>
<tr>
<td>macos-7_3-10.2</td>
<td>Apple - Cyrillic</td>
<td>1</td>
</tr>
<tr>
<td>ibm-437_P100-1995</td>
<td>DOS</td>
<td>1</td>
</tr>
<tr>
<td>ibm-850_P100-1995</td>
<td>DOS</td>
<td>1</td>
</tr>
<tr>
<td>ibm-851_P100-1995</td>
<td>DOS -Greek</td>
<td>1</td>
</tr>
<tr>
<td>Name</td>
<td>Typical Use</td>
<td>Character Length in Bytes</td>
</tr>
<tr>
<td>-----------------------</td>
<td>------------------------------</td>
<td>---------------------------</td>
</tr>
<tr>
<td>ibm-852_P100-1995</td>
<td>DOS</td>
<td>1</td>
</tr>
<tr>
<td>ibm-855_P100-1995</td>
<td>DOS - Cyrillic</td>
<td>1</td>
</tr>
<tr>
<td>ibm-856_P100-1995</td>
<td>DOS - Hebrew</td>
<td>1</td>
</tr>
<tr>
<td>ibm-857_P100-1995</td>
<td>DOS - Turkish</td>
<td>1</td>
</tr>
<tr>
<td>ibm-858_P100-1997</td>
<td>DOS</td>
<td>1</td>
</tr>
<tr>
<td>ibm-860_P100-1995</td>
<td>DOS - Portuguese</td>
<td>1</td>
</tr>
<tr>
<td>ibm-861_P100-1995</td>
<td>DOS - Icelandic</td>
<td>1</td>
</tr>
<tr>
<td>ibm-862_P100-1995</td>
<td>DOS - Hebrew</td>
<td>1</td>
</tr>
<tr>
<td>ibm-863_P100-1995</td>
<td>DOS - Canadian French</td>
<td>1</td>
</tr>
<tr>
<td>ibm-864_X110-1999</td>
<td>DOS - Arabic</td>
<td>1</td>
</tr>
<tr>
<td>ibm-865_P100-1995</td>
<td>DOS - Nordic</td>
<td>1</td>
</tr>
<tr>
<td>ibm-866_P100-1995</td>
<td>DOS - Russian</td>
<td>1</td>
</tr>
<tr>
<td>ibm-867_P100-1998</td>
<td>DOS - Hebrew</td>
<td>1</td>
</tr>
<tr>
<td>ibm-868_P100-1995</td>
<td>DOS</td>
<td>1</td>
</tr>
<tr>
<td>ibm-869_P100-1995</td>
<td>DOS - Greek</td>
<td>1</td>
</tr>
<tr>
<td>ibm-878_P100-1996</td>
<td>DOS - Russian</td>
<td>1</td>
</tr>
<tr>
<td>ibm-897_P100-1995</td>
<td>DOS - Japanese</td>
<td>1</td>
</tr>
<tr>
<td>ISClI,version=0</td>
<td>ISClI - Devanagari</td>
<td>3+</td>
</tr>
<tr>
<td>ISClI,version=1</td>
<td>ISClI - Bengali</td>
<td>3+</td>
</tr>
<tr>
<td>ISClI,version=2</td>
<td>ISClI - Gurmukhi</td>
<td>3+</td>
</tr>
<tr>
<td>ISClI,version=3</td>
<td>ISClI - Gujurati</td>
<td>3+</td>
</tr>
<tr>
<td>ISClI,version=4</td>
<td>ISClI - Orayi</td>
<td>3+</td>
</tr>
<tr>
<td>ISClI,version=5</td>
<td>ISClI - Tamil</td>
<td>3+</td>
</tr>
<tr>
<td>ISClI,version=6</td>
<td>ISClI - Telugu</td>
<td>3+</td>
</tr>
<tr>
<td>ISClI,version=7</td>
<td>ISClI - Kanada</td>
<td>3+</td>
</tr>
<tr>
<td>ISClI,version=8</td>
<td>ISClI - Malayalam</td>
<td>3+</td>
</tr>
<tr>
<td>ISO_2022.locale=ja,version=0</td>
<td>ISO2022 - Japanese</td>
<td>3+</td>
</tr>
</tbody>
</table>
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.

<table>
<thead>
<tr>
<th>Name</th>
<th>Typical Use</th>
<th>Character Length in Bytes</th>
</tr>
</thead>
<tbody>
<tr>
<td>ISO_2022,locale=ja,version=1</td>
<td>ISO2022 - Japanese</td>
<td>3+</td>
</tr>
<tr>
<td>ISO_2022,locale=ja,version=2</td>
<td>ISO2022 - Japanese</td>
<td>3+</td>
</tr>
<tr>
<td>ISO_2022,locale=ja,version=3</td>
<td>ISO2022 - Japanese</td>
<td>3+</td>
</tr>
<tr>
<td>ISO_2022,locale=ja,version=4</td>
<td>ISO2022 - Japanese</td>
<td>3+</td>
</tr>
<tr>
<td>ISO_2022,locale=ko,version=0</td>
<td>ISO2022 - Korean</td>
<td>3+</td>
</tr>
<tr>
<td>ISO_2022,locale=ko,version=1</td>
<td>ISO2022 - Korean</td>
<td>3+</td>
</tr>
<tr>
<td>ISO_2022,locale=zh,version=0</td>
<td>ISO2022 - Chinese</td>
<td>3+</td>
</tr>
<tr>
<td>ISO_2022,locale=zh,version=1</td>
<td>ISO2022 - Chinese</td>
<td>3+</td>
</tr>
<tr>
<td>LMBCS-1</td>
<td>Lotus Notes</td>
<td>1 to 2</td>
</tr>
<tr>
<td>LMBCS-11</td>
<td>Lotus Notes - Thai</td>
<td>1 to 2</td>
</tr>
<tr>
<td>LMBCS-16</td>
<td>Lotus Notes - Japanese</td>
<td>1 to 2</td>
</tr>
<tr>
<td>LMBCS-17</td>
<td>Lotus Notes - Korean</td>
<td>1 to 2</td>
</tr>
<tr>
<td>LMBCS-18</td>
<td>Lotus Notes - Chinese</td>
<td>1 to 2</td>
</tr>
<tr>
<td>LMBCS-19</td>
<td>Lotus Notes - Chinese</td>
<td>1 to 2</td>
</tr>
<tr>
<td>LMBCS-2</td>
<td>Lotus Notes - Greek</td>
<td>1 to 2</td>
</tr>
<tr>
<td>LMBCS-3</td>
<td>Lotus Notes - Hebrew</td>
<td>1 to 2</td>
</tr>
<tr>
<td>LMBCS-4</td>
<td>Lotus Notes - Arabic</td>
<td>1 to 2</td>
</tr>
<tr>
<td>LMBCS-5</td>
<td>Lotus Notes - Cyrillic</td>
<td>1 to 2</td>
</tr>
<tr>
<td>LMBCS-6</td>
<td>Lotus Notes - Central &amp; East Europe</td>
<td>1 to 2</td>
</tr>
<tr>
<td>LMBCS-8</td>
<td>Lotus Notes - Turkish</td>
<td>1 to 2</td>
</tr>
</tbody>
</table>
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:

<table>
<thead>
<tr>
<th>Data Type and Context</th>
<th>Result</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>Names of databases, tables, columns, and files during initialization.</td>
<td>Processing aborts.</td>
<td>Check the control code page.</td>
</tr>
<tr>
<td>User ID and password during connect processing.</td>
<td>Processing aborts.</td>
<td>Check the control code page.</td>
</tr>
<tr>
<td>SQL during Describe processing.</td>
<td>Processing aborts.</td>
<td>Check the SQL code page.</td>
</tr>
<tr>
<td>Column data during select processing.</td>
<td>Unrecognized characters are replaced with the substitution characters of the target code page which are hexadecimal values. Processing continues.</td>
<td>Check the data code page. Verify that the source data is described correctly using DTLDescribe columns.</td>
</tr>
<tr>
<td>Parameter data during delete, insert, select, and update processing.</td>
<td>Unrecognized characters are replaced with the substitution characters of the target code page which are hexadecimal values. Processing continues.</td>
<td>Check the data code page. Verify that the source and target data are described correctly using DTLDescribe columns.</td>
</tr>
</tbody>
</table>

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 1Fallback 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, 309**
- **PowerExchange ODBC Data Sources on Windows, 309**
- **PowerExchange ODBC Data Sources on Linux or UNIX, 324**
- **SQL Escape Sequences, 339**

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 the following PowerExchange ODBC drivers:

- **Informatica PowerExchange.** This driver accesses data and metadata for PowerExchange sources and targets.
- **Informatica PowerExchange Unicode.** 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.

Creating an ODBC Data Source on Windows

Use the ODBC Data Source Administrator to create a PowerExchange ODBC data source on Windows.

Windows 64-bit systems include a 32-bit version and a 64-bit version of the ODBC Data Source Administrator. To create a PowerExchange ODBC data source on a Windows 64-bit system, use the 32-bit version of the ODBC Data Source Administrator.

1. To start the ODBC Data Source Administrator, perform one of the following actions:
   - On a Windows 32-bit system, open the Control Panel and click Administrative Tools. Then double-click Data Sources (ODBC).
   - On a Windows 64-bit system, enter the following command at the command prompt:
     ```
     %windir%\SysWOW64\odbcad32.exe
     ```
     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:

<table>
<thead>
<tr>
<th>Data Source or Access Type</th>
<th>Properties Tab</th>
</tr>
</thead>
<tbody>
<tr>
<td>CAPX</td>
<td>CAPX Properties tab</td>
</tr>
<tr>
<td>CAPXRT</td>
<td>CAPXRT Properties tab</td>
</tr>
<tr>
<td>DB2 for i5/OS</td>
<td>AS/400 Properties tab</td>
</tr>
<tr>
<td>DB2 for z/OS</td>
<td>DB2/S390 Bulk Load Properties tab</td>
</tr>
</tbody>
</table>

7. Click OK.
   The ODBC data source appears in the System Data Sources list on the System DSN tab.

8. Click OK.

**RELATED TOPICS:**
- "General Tab" on page 314
- "AS/400 Properties Tab" on page 317
- "CAPX Properties Tab" on page 318
- "CAPXRT Properties Tab" on page 320
Accessing Multibyte Metadata with the PowerExchange ODBC Unicode Driver on Windows

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.

Windows 64-bit systems include a 32-bit version and a 64-bit version of the ODBC Data Source Administrator. To create a PowerExchange ODBC data source on a Windows 64-bit system, use the 32-bit version of the ODBC Data Source Administrator.

1. To start the ODBC Data Source Administrator, perform one of the following actions:
   - On a Windows 32-bit system, double-click Administrative Tools in the Control Panel. In the Administrative Tools window, double-click Data Sources (ODBC).
   - On a Windows 64-bit system, enter the following command at the command prompt:

     ```
     %windir%\SysWOW64\odbcad32.exe
     ```

     The ODBC Data Source Administrator appears.

2. On the System DSN tab, click Add.

   The Create New Data Source wizard appears.

3. 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 311

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.
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.

File Name
For the ADAUNLD data source type only, the name of the Adabas unload file.
Corresponding ODBC Parameter: DBQUAL1

Database
For the ADAUNLD, INFORMIX, MSSQL, or ODBC data source type only, the name of the database.
Corresponding ODBC Parameters: For the INFORMIX data source type, DBQUAL1. For the ADAUNLD, 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: DBQUAL2

**Application Name**

For the EMR data source type only, the application name used for the data extraction.

Corresponding ODBC Parameter: DBQUAL2

**Wait Time**

For the EMR, NRDB, or NRDB2 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

**Server**

For the INFORMIX data source type only, the server on which the database is located.

Corresponding ODBC Parameter: DBQUAL2

**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 310
- "PowerExchange ODBC Driver Data Source Parameters" on page 326
- "AS/400 Properties Tab" on page 317
- "CAPX Properties Tab" on page 318
- "CAPXRT Properties Tab" on page 320
- "DB2/S390 Bulk Load Properties Tab" on page 322
- "General Tab" on page 314

**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, SQLLEN4BYTES, 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.

- **SQLLEN4BYTES.** 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 is one of the following:

- **DES.** Data is encrypted by using Digital Encryption Standard (DES) algorithm.
- **RC2.** Data is encrypted by using the RSA security algorithm.

If you select an encryption option, you must select an encryption level in the **Level** list.

Corresponding ODBC Parameter: ENCRYPT, as follows:

- If you select the **DES** option, ENCRYPT=DES.
- If you select the **RC2** option, ENCRYPT=RC2.

Level

Required if you select the **DES** or **RC2** encryption option.

The encryption level.

The options are:

- **1.** Data is encrypted by using:
  - For DES: 56-bit key encryption
  - For RC2: 56-bit key encryption
- **2.** Data is encrypted by using:
  - For DES: Triple DES 164-bit key encryption
  - For RC2: 64-bit key encryption
- **3.** Data is encrypted by using:
  - For DES: Triple DES 164-bit key encryption
  - For RC2: 128-bit key encryption

Default is **1** if the **DES** or **RC2** 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 timeout and heartbeat interval overrides that apply to PowerExchange ODBC connections for bulk data movement or CDC sessions:

TCPIP_OP_TIMEOUT=network_operation_timeout
TCPIP_CON_TIMEOUT=connection_timeout
TCPIP_HB_INTERVAL=nnnnn

If you specify multiple overrides, use a comma (,) as a separator, for example:

TCPIP_OP_TIMEOUT=nnn,TCPIP_CON_TIMEOUT=nnn

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.

TCP_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.

TCP_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.

Corresponding ODBC Parameter: PWXOVERRIDES

Related Topics:
- "PowerExchange ODBC Driver Data Source Parameters" on page 326
- "AS/400 Properties Tab" on page 317
- "CAPX Properties Tab" on page 318
- "CAPXRT Properties Tab" on page 320
- "DB2/S390 Bulk Load Properties Tab" on page 322
- "PowerExchange Data Source Tab" on page 311

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: ISO\_LATION

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: OVR\_DBF

Related Topics:

- "PowerExchange ODBC Driver Data Source Parameters" on page 326
- "CAPX Properties Tab" on page 318
- "CAPXRT Properties Tab" on page 320
- "DB2/S390 Bulk Load Properties Tab" on page 322
- "General Tab" on page 314
- "PowerExchange Data Source Tab" on page 311

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 326
- "AS/400 Properties Tab" on page 317
- "CAPXRT Properties Tab" on page 320
- "DB2/S390 Bulk Load Properties Tab" on page 322
- "General Tab" on page 314
- "PowerExchange Data Source Tab" on page 311

**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**

Overwrites 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=(col1_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=(col1_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

IDMS LOGSID

For IDMS synchronous only. Overrides the instance value specified in the LOGSID statement in the dbmover.cfg configuration file, and the instance value specified in the PowerExchange capture registration.

Corresponding ODBC Parameter: LOGSID

DB2 UDB Database

For DB2 for Linux, UNIX, and Windows only. Required for EMR 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: XTRASHEMA

**RELATED TOPICS:**

- "PowerExchange ODBC Driver Data Source Parameters" on page 326
- "AS/400 Properties Tab" on page 317
- "CAPX Properties Tab" on page 318
- "DB2/S390 Bulk Load Properties Tab" on page 322
- "General Tab" on page 314
- "PowerExchange Data Source Tab" on page 311

**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 326
- "AS/400 Properties Tab" on page 317
- "CAPX Properties Tab" on page 318
- "CAPXRT Properties Tab" on page 320
- "General Tab" on page 314
- "PowerExchange Data Source Tab" on page 311

---

**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 326
- "PowerExchange ODBC Driver for DB2 for z/OS Bulk Parameters" on page 337

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
infranodb=INFA PWX850 NRDB
loaddbz=INFA PWX850 DB2LOAD
oracleunicode=INFA PWX850 UNICODE

[infadb2]
DRIVER=/usr/powerexchange/bin/libdtlodbc.sl
DESCRIPTION='Descriptive Text for DB2 Data Source section'
DBTYPE=db2
LOCATION=db2
DBQUAL1=dsn1

[infranodb]
DRIVER=/usr/powerexchange/bin/libdtlodbc.so
DESCRIPTION='Freeform text for nrdb section'
DBTYPE=nrdb
LOCATION=laptop

[infranodb2]
DRIVER=/usr/powerexchange/bin/libdtlodbc.so
DESCRIPTION='Freeform text for nrdb2 section'
DBTYPE=nrdb2
LOCATION=node1

[loaddb2]
DRIVER=/usr/1w/stprod310/libdtlodbc.so
DESCRIPTION=LoadDB2
DBTYPE=DB2
LOCATION=RemoteLOC (From DBMOVER.CFG)
DBQUAL1=SSID (target database SSID)
BULKLOAD=Y
JCLTEMPLATE=DB2LDCTL
JCLTEMPLATE=DB2LDJCL
DELETETEMPLFILES=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/vr850/libdtlodbcu.so
DESCRIPTION=
DATABASE=
DBTYPE=ORACLE
LOCATION=tortilla
DBQUAL1=TCP10UTF
MAXROWS=0
COMPRESS=N
ENCRT=N
CONFIRMWRITE=N
PACESIZE=0
INTERPRETASROWS=N
BULKLOAD=N
DELETETEMPFILES=0
LOADOPTIONS=0
LOCALCODEPAGE=41
MODETYPE=0
MDETIME=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 libdtlodbcinst.so or libdtlodbcinst.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 325, the oracleunicode data source is defined and the following parameters set:

```
DRIVER=/dt1ga2/v850/libdtlodbcu.so
LOCALCODEPAGE=41
```

**Related Topics:**

- "Example PowerExchange Data Source Specifications" on page 325

### 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:

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Default</th>
<th>Tokens</th>
<th>Usage</th>
</tr>
</thead>
</table>
| CAPXIMAGETYPE    | None    | BA, AI, or TU  | The capture image type for accessing PowerExchange CDC or live data. 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. |
| CAPXTYPE         | None    | SL or RS       | The type of extraction. 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.  
<p>|                  |         |                | - RS. Restart the extraction from the start point of the last extraction. This option enables you to run the same extraction multiple times.  |</p>
<table>
<thead>
<tr>
<th>Parameter</th>
<th>Default</th>
<th>Tokens</th>
<th>Usage</th>
</tr>
</thead>
</table>
| COMPATIBILITY    | None    | -      | The integration mask. Includes 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 by means of a JDBC-ODBC bridge connection to the PowerExchange ODBC driver. The ILMBRIDGE option enables the following options:  
  BINASCHAR, CLOSEDSTMREUSE, NOFKEYS, NOUNDERSCORE, SQLLEN4BYTES, 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.  
In addition, you can enable the following COMPATIBILITY 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.  
- SQLLEN4BYTES. 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. |
<table>
<thead>
<tr>
<th>Parameter</th>
<th>Default</th>
<th>Tokens</th>
<th>Usage</th>
</tr>
</thead>
<tbody>
<tr>
<td>COMPRESS</td>
<td>N</td>
<td>Y, N</td>
<td>Set this parameter to compress data. Default is N.</td>
</tr>
</tbody>
</table>
| 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:  
  - 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 PowerExchange Bulk Data Movement Guide.  
  - 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. |
<p>| DATABASE      | -       | -      | For DB2 for Linux, UNIX, and Windows only. Required for EMR only. Overrides the database connection value specified in the PowerExchange extraction map.                                                      |</p>
<table>
<thead>
<tr>
<th>Parameter</th>
<th>Default</th>
<th>Tokens</th>
<th>Usage</th>
</tr>
</thead>
</table>
| DB2DEGREE     | None    | Any string of text up to 30 characters | 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: `SET CURRENT DEGREE`  
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.  
For z/OS only: Any text following the equals sign is taken as text to follow a `'SET CURRENT DEGREE =` statement.  
For example, specifying `DB2DEGREE=ANY` runs the statement `''SET CURRENT DEGREE = 'ANY' ''`, and would set DB2 to be able to use parallel processing.  
For i5/OS only: Any text following the equals sign is taken as text to follow a `'CHGQRYA DEGREE='` statement.  
For example, specifying `DB2DEGREE=SYSTEM` runs 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)*NBRTASKS1' '' statement. Thus a `'CHGQRYA DEGREE=('NBRTASKS1)''' statement can be generated with either a `DB2DEGREE=1` or `DB2DEGREE=''NBRTASKS1''` 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. |</p>
<table>
<thead>
<tr>
<th>Parameter</th>
<th>Default</th>
<th>Tokens</th>
<th>Usage</th>
</tr>
</thead>
<tbody>
<tr>
<td>DTLCONN_OVR</td>
<td>None</td>
<td>Not applicable</td>
<td>The name in the CAP_CONNECTION statement in the DBMOVER configuration file that points to the relevant data source.</td>
</tr>
<tr>
<td>DTLIMGOV</td>
<td>N</td>
<td>Y or N</td>
<td>For DB2390IMG only. 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. You can override this behavior by setting the DTLIMGOV parameter to Y. This setting enables inconsistent image copies to be read.</td>
</tr>
</tbody>
</table>
| ENCRYPT        | N       | N, Y, DES, or RC2 | The encryption option. The options are:  
- 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.  
- DES. Data is encrypted by using Digital Encryption Standard (DES) algorithm.  
- RC2. Data is encrypted by using the RSA security algorithm.  
If you specify an encryption option, you must also specify an encryption level in the ENCRYPTLEVEL parameter. |
| ENCRYPTLEVEL   | -       | 1, 2 or 3 | The encryption level. Required if the ENCRYPT parameter is set to DES or RC2.  
The options are:  
1. Data is encrypted by using:  
   - For DES: 56-bit key encryption  
   - For RC2: 56-bit key encryption  
2. Data is encrypted by using:  
   - For DES: Triple DES 164-bit key encryption  
   - For RC2: 64-bit key encryption  
3. Data is encrypted by using:  
   - For DES: Triple DES 164-bit key encryption  
   - For RC2: 128-bit key encryption |
| INTERPRETASROWS| N       | Y, N   | Indicates whether to interpret the pacing size in rows or kilobytes. The options are:  
- Y. Interpret the pacing size in rows.  
- N. Interpret the pacing size in kilobytes.  
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. |</p>
<table>
<thead>
<tr>
<th>Parameter</th>
<th>Default</th>
<th>Tokens</th>
<th>Usage</th>
</tr>
</thead>
<tbody>
<tr>
<td>JRNL</td>
<td>-</td>
<td>LIBRARY/JOURNAL</td>
<td>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. For example: STQA/NEWJOURNAL</td>
</tr>
<tr>
<td>LIBRARYLIST</td>
<td>Maximum of 34 libraries. Each library name is a maximum length of ten characters.</td>
<td></td>
<td>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.</td>
</tr>
<tr>
<td>LOCALCODEPAGE</td>
<td>None</td>
<td>-</td>
<td>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 iccheck.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: iccheck.exe &gt; iculist.txt</td>
</tr>
<tr>
<td>LOCATION</td>
<td>None</td>
<td>None</td>
<td>The location of the data source. The location is defined by a NODE statement in the dbmover.cfg configuration file on the local system.</td>
</tr>
<tr>
<td>LOGSID</td>
<td>-</td>
<td>-</td>
<td>For IDMS synchronous only. Overrides the instance value specified in the LOGSID statement in the dbmover.cfg configuration file, and the instance value specified in the PowerExchange capture registration.</td>
</tr>
<tr>
<td>MAXROWS</td>
<td>0</td>
<td>0 to 2147483647</td>
<td>The maximum number of rows for retrieval. Default is 0, which indicates unlimited rows.</td>
</tr>
<tr>
<td>MQGETOPT</td>
<td>R</td>
<td>B or R</td>
<td>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.</td>
</tr>
<tr>
<td>Parameter</td>
<td>Default</td>
<td>Tokens</td>
<td>Usage</td>
</tr>
<tr>
<td>-----------------</td>
<td>---------</td>
<td>--------</td>
<td>-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>NOUPDATECDEP</td>
<td>N</td>
<td>Y or N</td>
<td>Indicates whether database row test output is to be added to the PowerExchange CDC audit trail. The options are:</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>- Y. Omit the database row test output from the PowerExchange CDC audit trail for the application.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>- N. Include the database row test output in the PowerExchange CDC audit trail for the application.</td>
</tr>
<tr>
<td>ORACOLL</td>
<td>-</td>
<td>-</td>
<td>Overrides the value specified in the ORACOLL parameter in the ORCL 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.</td>
</tr>
<tr>
<td>ORACONN</td>
<td>-</td>
<td>-</td>
<td>Overrides the Oracle connection string value in the third positional parameter in the ORACLEID statement in the dbmover.cfg configuration file. For example:</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>ORACLEID=(coll_id, oracle_sid, connect_string, cap_connect_string)</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>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.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>If you specify an ORACONN value, but no ORAINST value, Oracle capture uses the Oracle connection string value specified in the ORACLEID statement in the dbmover.cfg file.</td>
</tr>
<tr>
<td>ORAINST</td>
<td>-</td>
<td>-</td>
<td>Overrides the Oracle instance value in the second positional parameter in the ORACLEID statement in the dbmover.cfg configuration file. For example:</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>ORACLEID=(coll_id, oracle_sid, connect_string, cap_connect_string)</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>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.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>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.</td>
</tr>
<tr>
<td>Parameter</td>
<td>Default</td>
<td>Tokens</td>
<td>Usage</td>
</tr>
<tr>
<td>-------------</td>
<td>---------</td>
<td>--------</td>
<td>-------</td>
</tr>
<tr>
<td>ORASHEMA</td>
<td>-</td>
<td>-</td>
<td>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.</td>
</tr>
<tr>
<td>OVRDBF</td>
<td>-</td>
<td>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. Space-delimited list of file overrides in the following format: <code>filename/newlibrary/newfilename/newmembername</code> Alternatively, you can use the following format: <code>filename/newlibrary/newfilename/</code> In this format, the member defaults to <code>*FIRST</code>. When you use this override, any occurrence of the specified file name in a SQL statement is overridden with the <code>newlibrary/newfilename/newmembername</code> combination, regardless of whether the file is qualified or not. This includes any files that are specified in the LIBRARYLIST override. <strong>Note:</strong> 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.</td>
<td></td>
</tr>
<tr>
<td>PACESIZE</td>
<td>Value of the API <code>rows_requested</code> parameter. Numeric</td>
<td>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. For maximum performance, set this parameter to 0.</td>
<td></td>
</tr>
<tr>
<td>Parameter</td>
<td>Default</td>
<td>Tokens</td>
<td>Usage</td>
</tr>
<tr>
<td>-----------------</td>
<td>---------</td>
<td>--------</td>
<td>-------</td>
</tr>
</tbody>
</table>
| PWXOVERRIDES    | None    | -      | Timeout and heartbeat overrides that apply to PowerExchange ODBC connections for bulk data movement and CDC sessions:  
|                 |         |        | - 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.  
|                 |         |        | - TCP_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 the following syntax:  
|                 |         |        | PWXOVERRIDES=TCPIP_OP_TIMEOUT=network_operation_timeout  
|                 |         |        | PWXOVERRIDES=TCPIP_CON_TIMEOUT=connection_timeout  
|                 |         |        | PWXOVERRIDES=TCPIP_HB_INTERVAL=n.nn  
|                 |         |        | If you specify multiple overrides, use a semicolon (;) as a separator, for example:  
|                 |         |        | PWXOVERRIDES=TCPIP_OP_TIMEOUT=n.nn;TCP_CON_TIMEOUT=n.nn |
| REJECTFILE      | None    | Up to 384 characters | Available only if the CONFIRMWRITE parameter is set to T, which enables asynchronous with fault tolerance mode.  
|                 |         |        | The location and file name of the reject file that is used during asynchronous write operations. Enter a value of PWXDISABLE to disable reject file logging.  
|                 |         |        | For more information, see the PowerExchange Bulk Data Movement Guide. |
| STOPONERRORS    | 0       | 0 to 2147483647 | Available only if the CONFIRMWRITE parameter is set to T, which enables asynchronous with fault tolerance mode.  
|                 |         |        | The number of non-fatal errors that are allowed in the write phase before processing is terminated.  
|                 |         |        | For more information, see the PowerExchange Bulk Data Movement Guide. |
### Parameter Description

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Default</th>
<th>Tokens</th>
<th>Usage</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>UAP</strong></td>
<td>None</td>
<td>-</td>
<td>For DB2 for i5/OS data sources only. Overrides the fully qualified library and journal name specified in the PowerExchange capture registration. For example: <code>STQA/NEWJOURNAL</code></td>
</tr>
<tr>
<td><strong>WAITTIME</strong></td>
<td>-</td>
<td>0, 2 to 86399, 86400</td>
<td>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.</td>
</tr>
<tr>
<td><strong>XTRASHEMA</strong></td>
<td>-</td>
<td>Up to 128 characters with no spaces allowed.</td>
<td>Overrides the schema defined in the extraction map.</td>
</tr>
</tbody>
</table>

---

The following table describes the DBQUAL1 parameter:

<table>
<thead>
<tr>
<th>Tokens</th>
<th>DBType</th>
</tr>
</thead>
<tbody>
<tr>
<td>Filename</td>
<td>ADAUNLD</td>
</tr>
<tr>
<td>i5/OS Library/Filename</td>
<td>CAPX and CAPX/RT</td>
</tr>
<tr>
<td>SSID or DBName</td>
<td>DB2, DB2400C, and DB2UDB</td>
</tr>
<tr>
<td>Not applicable</td>
<td>DB2390IMG</td>
</tr>
<tr>
<td>Library/Filename</td>
<td>EMR for i5/OS</td>
</tr>
<tr>
<td>Not applicable</td>
<td>IMSUNLD</td>
</tr>
<tr>
<td>Database</td>
<td>INFORMIX</td>
</tr>
<tr>
<td>DSN</td>
<td>MSSQL</td>
</tr>
<tr>
<td>Not applicable</td>
<td>NRDB and NRDB2</td>
</tr>
<tr>
<td>SQL ‘Net Name</td>
<td>ORACLE</td>
</tr>
<tr>
<td>Sub-system</td>
<td>Teradata</td>
</tr>
</tbody>
</table>
The following table describes the DBQUAL2 parameter:

<table>
<thead>
<tr>
<th>Tokens</th>
<th>DBType</th>
</tr>
</thead>
<tbody>
<tr>
<td>Database</td>
<td>ADAUNLD</td>
</tr>
<tr>
<td>Application name</td>
<td>CAPX and CAPX/RT</td>
</tr>
<tr>
<td>Not applicable</td>
<td>DB2, DB2400C, and DB2UDB</td>
</tr>
<tr>
<td>SSID</td>
<td>DB2390IMG</td>
</tr>
<tr>
<td>Not applicable</td>
<td>IMSUNLD</td>
</tr>
<tr>
<td>Server</td>
<td>INFORMIX</td>
</tr>
<tr>
<td>Database</td>
<td>MSSQL</td>
</tr>
<tr>
<td>Not applicable</td>
<td>NRDB and NRDB2</td>
</tr>
<tr>
<td>Not applicable</td>
<td>ORACLE</td>
</tr>
<tr>
<td>Any supported data source, such as DB2</td>
<td>Access method for file or database.</td>
</tr>
</tbody>
</table>

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:

<table>
<thead>
<tr>
<th>Field</th>
<th>Type</th>
<th>Size</th>
<th>Valid Values</th>
<th>Comments</th>
</tr>
</thead>
<tbody>
<tr>
<td>BULKLOAD</td>
<td>Char</td>
<td>1</td>
<td>Y or N</td>
<td>Enables or disables DB2 bulk load processing. If you specify Y, you can configure the following parameters.</td>
</tr>
<tr>
<td>CTLTEMPLATE</td>
<td>Char</td>
<td>8</td>
<td>User-defined</td>
<td>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.</td>
</tr>
<tr>
<td>DELETETEMPFILES</td>
<td>Numeric</td>
<td>int</td>
<td>1, 2, 3, or 4</td>
<td>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</td>
</tr>
<tr>
<td>Field</td>
<td>Type</td>
<td>Size</td>
<td>Valid Values</td>
<td>Comments</td>
</tr>
<tr>
<td>------------</td>
<td>------</td>
<td>------</td>
<td>--------------</td>
<td>----------</td>
</tr>
</tbody>
</table>
| 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). |
| MODETIME   | Numeric | int | 1, 2, 3, or 4 | 1 = WAIT  
Synchronous. Report results at end. Applicable only if MODETYPE=1 or MODETYPE=3 is specified.  
2 = NOWAIT  
Asynchronous. Returns OK when job is submitted. Applicable only if MODETYPE=3 or MODETYPE=2 is specified.  
3 = TIMED  
Synchronous. Wait a specified number of seconds. Applicable only if MODETYPE=3 is specified.  
4 = DATAONLY  
Only creates the data DB2 Loader file. Applies to all mode type values, but typically used with NOSUBMIT, MODETYPE=2. |
| MODETYPE   | Numeric | int | 1, 2, or 3   | 1 = TASK  
Submitted as a subtask of the PowerExchange Listener.  
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.  
3 = JOB  
Submitted to the internal reader as a job. |
| 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. |
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} d@roboral

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 340
- "(DTLADAPWD= ) Escape Sequence" on page 340
- "(DTLAPP= ) Escape Sequence" on page 340
- "(DTLCONNOVR= ) Escape Sequence" on page 340
- "(DTLCONFWRITE=N|Y|T) Escape Sequence" on page 340
- "(DTLDATAPWD= ) Escape Sequence" on page 341
- "(DTLDB2DEGREE= ) Escape Sequence" on page 341
- "(DTLDSN= ) Escape Sequence" on page 341
- "(DTEVENTTB= ) Escape Sequence" on page 342
- "(DTLIMTYPE=BA|AI|ITU) Escape Sequence" on page 342
- "(DTLIMGOV=Y|N) Escape Sequence" on page 342
- "(DTLINSMODE=LOAD|UPDATE) Escape Sequence" on page 342
- "(DTLJRNL= ) Escape Sequence" on page 342

<table>
<thead>
<tr>
<th>Field</th>
<th>Type</th>
<th>Size</th>
<th>Valid Values</th>
<th>Comments</th>
</tr>
</thead>
<tbody>
<tr>
<td>SPACE</td>
<td>Char</td>
<td>1</td>
<td>T = Tracks</td>
<td>z/OS allocation of space.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>C = Cylinders</td>
<td></td>
</tr>
<tr>
<td>TIME</td>
<td>Numeric</td>
<td>int</td>
<td>0 to 99999</td>
<td>A value from 0 to 99999, as follows:</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>- 0. No wait.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>- 99999. Wait.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>- Any other value. The number of seconds to wait. Only used in conjunction with MODE=(JOB,TIMED).</td>
</tr>
</tbody>
</table>
• “{DTL LIBRARYLIST=} Escape Sequence” on page 342
• “{DTL NUPDATECDEP=Y|N} Escape Sequence” on page 343
• “{DTL ORACOLL=} Escape Sequence” on page 343
• “{DTL ORACON=} Escape Sequence” on page 343
• “{DTL ORAINST=} Escape Sequence” on page 343
• “{DTL ORAS CHEMA=} Escape Sequence” on page 343
• “{DTL OVRDBF=} Escape Sequence” on page 343
• “{DTL REJECTFILE=} Escape Sequence” on page 344
• “{DTL SESSID=} Escape Sequence” on page 344
• “{DTL STOPERRORS =} Escape Sequence” on page 344
• “{DTLSTRIPORDERBY=Y|N} Escape Sequence” on page 344
• “{DTL TIMEOUT=} Escape Sequence” on page 344
• “{DTL UDBFB=} Escape Sequence” on page 344
• “{DTL UPDELEINS=Y} Escape Sequence” on page 344
• “{DTL WORKERS=} Escape Sequence” on page 344
• “{DTL XTRAS CHEMA=} Escape Sequence” on page 345
• “{DTL XTYPE=RS|SL} Escape Sequence” on page 345

{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.

{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.
{DTLDATAPWD= } 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}.

- 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.

{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.
{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.

{DTLORASCHHEMA= } 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.

**{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.
PowerExchange Datatypes and Conversion Matrix

This chapter includes the following topics:

- PowerExchange Datatype Conversions, 346
- ODBC and PowerExchange Datatype Equivalencies, 347

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:

<table>
<thead>
<tr>
<th>From</th>
<th>To BIN</th>
<th>To CHAR</th>
<th>To DATE</th>
<th>To DBL</th>
<th>To FLT</th>
<th>To STRING</th>
<th>To TIME</th>
<th>To TIMESTAMP</th>
<th>To VARBIN</th>
<th>To VARCHAR</th>
</tr>
</thead>
<tbody>
<tr>
<td>BIN</td>
<td>X</td>
<td>X</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>X</td>
<td>-</td>
<td>-</td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td>DATE</td>
<td>-</td>
<td>X</td>
<td>X</td>
<td>-</td>
<td>-</td>
<td>X</td>
<td>-</td>
<td>X</td>
<td>-</td>
<td>X</td>
</tr>
<tr>
<td>DBL</td>
<td>-</td>
<td>X</td>
<td>-</td>
<td>X</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>X</td>
</tr>
<tr>
<td>FLT</td>
<td>-</td>
<td>X</td>
<td>-</td>
<td>X</td>
<td>X</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>X</td>
</tr>
<tr>
<td>TIME</td>
<td>-</td>
<td>X</td>
<td>-</td>
<td>-</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>-</td>
<td>X</td>
<td>-</td>
</tr>
<tr>
<td>TIMESTAMP</td>
<td>-</td>
<td>X</td>
<td>X</td>
<td>-</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>-</td>
<td>X</td>
<td>-</td>
</tr>
<tr>
<td>VARBIN</td>
<td>X</td>
<td>X</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>X</td>
<td>-</td>
<td>-</td>
<td>X</td>
<td>X</td>
</tr>
</tbody>
</table>
The following table lists the supported conversions for the CHAR, NUMCHAR, NUMxx, PACKED, STRING, UPACKED, UZONED, VARCHAR, and ZONED datatypes:

<table>
<thead>
<tr>
<th>From</th>
<th>To CHAR</th>
<th>To DBL</th>
<th>To NUMCHAR</th>
<th>To NUMxx</th>
<th>To PACKED</th>
<th>To STRING</th>
<th>To VARCHAR</th>
<th>To ZONED</th>
<th>UZONED</th>
</tr>
</thead>
<tbody>
<tr>
<td>CHAR</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td>NUMCHAR</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td>NUMxx</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td>PACKED</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td>STRING</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td>UPACKED</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td>UZONED</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td>VARCHAR</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td>ZONED</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
</tr>
</tbody>
</table>

ODBC and PowerExchange Datatype Equivalencies

The following table shows the ODBC and PowerExchange datatype equivalencies:

<table>
<thead>
<tr>
<th>ODBC SQL Datatype</th>
<th>ODBC SQL C Datatype</th>
<th>PowerExchange Datatype</th>
</tr>
</thead>
<tbody>
<tr>
<td>SQL_BINARY</td>
<td>-</td>
<td>DTLNET_BIN</td>
</tr>
<tr>
<td>SQL_BIT</td>
<td>SQL_C_BIT</td>
<td>DTLNET_BIT</td>
</tr>
<tr>
<td>SQL_CHAR</td>
<td>-</td>
<td>DTLNET_CHAR</td>
</tr>
<tr>
<td>SQL_DATE</td>
<td>SQL_C_DATE</td>
<td>DTLNET_DATE</td>
</tr>
<tr>
<td>SQL_DECIMAL</td>
<td>-</td>
<td>DTLNET_MONEY</td>
</tr>
<tr>
<td>SQL_DECIMAL</td>
<td>SQL_C_ULONG</td>
<td>DTLNET_NUM32U</td>
</tr>
<tr>
<td>SQL_DECIMAL</td>
<td>-</td>
<td>DTLNET_NUM64</td>
</tr>
<tr>
<td>SQL_DECIMAL</td>
<td>-</td>
<td>DTLNET_NUM64U</td>
</tr>
<tr>
<td>SQL_DECIMAL</td>
<td>-</td>
<td>DTLNET_NUMCHAR</td>
</tr>
<tr>
<td>SQL_DECIMAL</td>
<td>-</td>
<td>DTLNET_PACKED</td>
</tr>
<tr>
<td>ODBC SQL Datatype</td>
<td>ODBC SQL C Datatype</td>
<td>PowerExchange Datatype</td>
</tr>
<tr>
<td>-------------------</td>
<td>--------------------</td>
<td>------------------------</td>
</tr>
<tr>
<td>SQL_DECIMAL</td>
<td>-</td>
<td>DTLNET_UPACKED</td>
</tr>
<tr>
<td>SQL_DECIMAL</td>
<td>-</td>
<td>DTLNET_UZONED</td>
</tr>
<tr>
<td>SQL_DECIMAL</td>
<td>-</td>
<td>DTLNET_ZONED</td>
</tr>
<tr>
<td>SQL_DOUBLE</td>
<td>SQL_C_DOUBLE</td>
<td>DTLNET_DOUBLE</td>
</tr>
<tr>
<td>SQL_INTEGER</td>
<td>SQL_C_USHORT</td>
<td>DTLNET_NUM16U</td>
</tr>
<tr>
<td>SQL_INTEGER</td>
<td>SQL_C_LONG and SQL_C_SLONG</td>
<td>DTLNET_NUM32</td>
</tr>
<tr>
<td>SQL_REAL</td>
<td>SQL_C_FLOAT</td>
<td>DTLNET_FLOAT</td>
</tr>
<tr>
<td>SQL_SMALLINT</td>
<td>SQL_C_SHORT and SQL_C_SSHORT</td>
<td>DTLNET_NUM16</td>
</tr>
<tr>
<td>SQL_SMALLINT</td>
<td>SQL_C_UTINYINT</td>
<td>DTLNET_NUM8U</td>
</tr>
<tr>
<td>SQL_TIME</td>
<td>SQL_C_TIME</td>
<td>DTLNET_TIME</td>
</tr>
<tr>
<td>SQL_TIMESTAMP</td>
<td>SQL_C_TIMESTAMP</td>
<td>DTLNET_TIMESTAMP</td>
</tr>
<tr>
<td>SQL_TINYINT</td>
<td>SQL_C_TINYINT and SQL_C_STINYINT</td>
<td>DTLNET_NUM8</td>
</tr>
<tr>
<td>SQL_VARBINARY</td>
<td>SQL_C_BINARY</td>
<td>DTLNET_VARBIN</td>
</tr>
<tr>
<td>SQL_VARCHAR</td>
<td>SQL_C_CHAR</td>
<td>DTLNET_STRING</td>
</tr>
<tr>
<td>SQL_VARCHAR</td>
<td>-</td>
<td>DTLNET_VARCHAR</td>
</tr>
</tbody>
</table>
This appendix includes the following topic:

- **Time Stamps That Are Reported in the DTL__CAPXTIMESTAMP Field by Data Source, 349**

### 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 and for PowerExchange Oracle CDC with LogMiner sources, 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:

<table>
<thead>
<tr>
<th>Data Source Type</th>
<th>Time Stamp Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>Adabas</td>
<td>The HDDATE time stamp from the PLOG block header, which indicates when the block was written. <strong>Note:</strong> 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.</td>
</tr>
<tr>
<td>Datacom table-based CDC</td>
<td>The Coordinated Universal Time (UTC) time or local time when the change record was written to the Datacom LXX log. The <code>LOCAL_TIME</code> parameter in the ECCR configuration member, <code>ECCRDCMP</code>, controls whether the UTC or local time is used.</td>
</tr>
<tr>
<td>DB2 for i5/OS</td>
<td>An i5/OS journal time stamp that reflects when the change was recorded in the journal.</td>
</tr>
</tbody>
</table>
### Data Source Type

<table>
<thead>
<tr>
<th>Data Source Type</th>
<th>Time Stamp Type</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>DB2 for z/OS</strong></td>
<td>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.</td>
</tr>
<tr>
<td><strong>IDMS</strong></td>
<td>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.</td>
</tr>
<tr>
<td><strong>IMS log-based CDC</strong></td>
<td>The time at which the change was recorded in the IMS logs.</td>
</tr>
<tr>
<td><strong>IMS synchronous CDC</strong></td>
<td>The time at which the change occurred.</td>
</tr>
<tr>
<td><strong>Oracle CDC with LogMiner</strong></td>
<td>The time stamp of the change on the source database, as recorded in the redo logs. This time reflects the local time zone.</td>
</tr>
<tr>
<td><strong>Batch VSAM and CICS/VSAM</strong></td>
<td>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.</td>
</tr>
</tbody>
</table>

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:

<table>
<thead>
<tr>
<th>Data Source Type</th>
<th>Time Stamp Type</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>DB2 for Linux, UNIX, and Windows</strong></td>
<td>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.</td>
</tr>
<tr>
<td><strong>Microsoft SQL Server</strong></td>
<td>The time at which the change was written to the distribution database.</td>
</tr>
<tr>
<td><strong>PowerExchange Express CDC for Oracle</strong></td>
<td>The time stamp type is controlled by the TIME_STAMP_MODE parameter setting in the OPTIONS statement of the Express CDC configuration file.</td>
</tr>
<tr>
<td></td>
<td>- 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.</td>
</tr>
<tr>
<td></td>
<td>- If you specify COMMITTIME, PowerExchange reports the time stamp of the transaction commit on the source database.</td>
</tr>
<tr>
<td></td>
<td>- If you specify BEGINTIME, PowerExchange reports the time stamp of the begin UOW log record.</td>
</tr>
</tbody>
</table>
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 352.

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 352.

Agent
See PowerExchange Agent on page 363.

APF
See authorized program facility (APF) on page 352.

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 368.

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 368.

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 355 and real-time extraction mode on page 366.

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 351.

BMP

See Batch Message Processing (BMP) on page 352.

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 355.
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 datalib 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 364.

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 366.

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 354 and PowerExchange Change Data Capture (CDC) on page 363.
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 351.

Change Collector
A PowerExchange component for Datacom and IDMS sources that captures changes as they occur and passes the changes to the Change Controller. The Change Controller passes the changes to a data space that stores them until they are transmitted to the PowerExchange Logger. For Datacom sources only, you can use the more efficient direct-log-write method of capturing changes. In this case, the Change Collector passes the changes directly to the PowerExchange Logger. The Datacom Change Collector runs in the Datacom MUF address space, and the IDMS Change Collector runs in the IDMS CV address space. See also Change Controller on page 354 and Log Feeder on page 361.

Change Controller
A PowerExchange component for Datacom and IDMS sources that manages the capture registrations for the Change Collector. For IDMS sources and for any Datacom sources for which you do not use the direct-log-write method of change capture, the Change Controller also stores captured changes in an interim data space. The Log Feeder will get the changes from this data space. The Change Controller runs in an address space that is separate from the Change Collector address space. See also Change Controller on page 354 and Log Feeder on page 361.

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 355 and PowerExchange Change Data Capture (CDC) on page 363.

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 or for the PowerExchange Logger on Linux, UNIX, or Windows. This information is used during a warm start to properly resume PowerExchange Condense or PowerExchange Logger 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 364.

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 364.

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 352 and real-time extraction mode on page 366.

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 354 and PowerExchange Change Data Capture (CDC) on page 363.

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 367.
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.

Direct Log Write
The recommended method of capturing changes for Datacom sources. With this method, the Datacom Change Collector captures changes as they occur and passes them directly to the PowerExchange Logger. The capture registrations that the Change Collector uses are managed by the Change Controller. This direct-log-write method is more efficient than the older, alternative method that uses the Change Controller to store the changes in an interim data space and the Log Feeder to pass the changes from the data space to the PowerExchange Logger. See also Change Collector on page 354, Change Controller on page 354, and Log Feeder on page 361.

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 DTLUOGC utilities
PowerExchange utilities for updating the IDMS Log Catalog, which is used for IDMS log-based change data capture. DTLULCAT prepares the input for DTLUOGC. DTLUOGC 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 359.
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 355.
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 358.

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 355.

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 352, continuous extraction mode on page 355, and real-time extraction mode on page 366.

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 363.

**group definition file**

See *PowerExchange Condense group definition file* on page 364 or *PowerExchange Logger group definition file* on page 364.

**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 360.

**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 364.

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 Feeder
A PowerExchange component for Datacom and IDMS sources that reads captured changes from the data space in which the Change Controller stores them and passes the changes to PowerExchange Logger log files. The Log Feeder runs an address space that is separate from those of the other change-data-capture components. If you use the direct-log-write method of capturing changes for Datacom sources, the Log Feeder is not used for change data capture. See also Change Collector on page 354 and Change Controller on page 354.

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.

log-based capture
An asynchronous method of reading change data from log files.

Logger
See PowerExchange Logger for Linux, UNIX, and Windows on page 364 or PowerExchange Logger for MVS on page 364.

logger token
See restart tokens on page 367.

LRAPI
See Log Read API (LRAPI) on page 361.

LWAPI
See Log Write API (LWAPI) on page 361.

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 364.

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 367.

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 367.

 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 362.

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 364.

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 360.

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 MVS**
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 pwxccl.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.

**PWXCCL 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 364.

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 pwxccl.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.

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.

**PWXUSSL utility**

A PowerExchange utility that you use to generate reports about SSL libraries and certificates on Linux, UNIX, and Windows.

You can generate the following reports:

• Certificate report. Reports information from a certificate chain file. The report can include multiple certificates in a PEM chain file.
• Ciphers report. Reports 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.
• Version report. Reports the version of OpenSSL that was used to build the cryptographic library. On Linux and UNIX, the cryptographic library file is named libpmcrypto. 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 352` and `continuous extraction mode on page 355`.

**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 353`.
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 358**.

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 362**.

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 362**.

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 363**.

sequence token
See **restart tokens on page 367**.
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 362.

simple table
A table in a data map that is based on a single record.

source
See data source on page 356.

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 Datacom, IDMS synchronous, IMS synchronous, batch VSAM, and CICS/VSAM sources. IMS and IDMS log-based sources use asynchronous capture. See also asynchronous capture on page 352.

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 352.

tag
See registration tag on page 367.

target
See data target on page 356.

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 368.

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.

**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.

**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 355*.

**XRF**

See *cross-system coupling facility (XCF) on page 355*.
INDEX

A
ABEND_SW statement
DBMOVER configuration file 34
accessing
multibyte metadata with ODBC 326
ADA_L3_ALLOW statement
DBMOVER configuration file 35
Adabas
security 218
Adabas DBMOVER statements 23
ADABAS_DEFAULT statement
DBMOVER configuration file 36
ADABAS_PREFIX statement
DBMOVER configuration file 36
ADABASCODEPAGE statement
DBMOVER configuration file 37
ADAOPT statement
DBMOVER configuration file 38
ADAPREFETCH statement
DBMOVER configuration file 39
ADASTATS statement
DBMOVER configuration file 39
ADAUSER statement
DBMOVER configuration file 40
adding
custom ICU code pages 283
user-defined ICU-compatible code pages 282
alternative log data sets on z/OS size of 164
alternative log files 164
alternative log format 163
APPBUFSIZE statement
DBMOVER configuration file 40
APPBUFSIZEEDYN statement
DBMOVER configuration file 41
AS/400 Properties tab
in PowerExchange Data Source wizard 317
AS400EVENTMSGQ statement
DBMOVER configuration file 41
AS400USRJRNCODE statement
DBMOVER configuration file 42
AS4J CAPI_CONNECTION parameters
parameters and syntax 44
AT-TLS
adding a rule 235
overview 235
policy file for PowerExchange 237
setting up infrastructure 235
authorizing
user credentials (Linux and UNIX) 205
user credentials (Windows) 205
AUTHGROUP statement
sign-on file 226
authorizing (continued)
users to run infacmd pwxcmd commands (Linux, UNIX, and Windows) 204
users to run PowerExchange Listener commands (i5/OS) 202
users to run PowerExchange Listener commands (Windows) 204
users to run pwxcmd commands 135, 195
users to run pwxcmd commands (i5/OS) 202
users to run pwxcmd commands (Linux, UNIX, and Windows) 204
users to run pwxcmd commands (z/OS) 221

B
BS statement
DBMOVER configuration file 43

C
CA certificate, creating 240
CA IDMS/DB statement
DBMOVER configuration file 24
CAPI connection statements
AS4J parameters 44
CAPX parameters 49
LRAP parameters 51
MSQL parameters 53
multiple 166
ORAD parameters 56
ORCL parameters 58
overview 166
UDB parameters 64
UOWC parameters 68
CAPI statements
order of precedence 168
CAPI_CONN_NAME statement
DBMOVER configuration file 43
CAPI_CONNECTION - AS4J statement
DBMOVER configuration file 44
CAPI_CONNECTION - CAPX statement
DBMOVER configuration file 49
CAPI_CONNECTION - LRAP statement
DBMOVER configuration file 51
CAPI_CONNECTION - MSQL statement
DBMOVER configuration file 53
CAPI_CONNECTION - ORAD statement
DBMOVER configuration file 56
CAPI_CONNECTION - ORCL statement
DBMOVER configuration file 58
CAPI_CONNECTION - UDB statement
DBMOVER configuration file 64
CAPI_CONNECTION - UOWC statement
DBMOVER configuration file 68
CAPI_SRC_DFLT statement
DBMOVER configuration file 73
DBMOVER statements (continued)

CONVCHAR 79
CPX_DIR 80
CREDENTIALS_CASE 80
DATAMAP_SERVER 81
DATERANGE 82
DB2_BIN_AS_CHAR 82
DB2_BIN_CODEPAGE 83
DB2_ERRORFILE 83
DB2CODEPAGE 84
DB2DEF_ENCODING 88
DB2ID 89
DB2PLAN 89
DECPOINT 90
DEFAULTCHAR 90
DEFAULTDATE 91
DISABLE_PARTITIONS 91
DISP 91
DM_RESOURCE 92
DM_SUBTASK 93
DMX_DIR 96
DMXCACHE_DELETEECSA 94
DMXCACHE_MAX_MEMORY_MB 94
DMXCACHE_MULTIPLEJOBS 95
DLMSG_CODEPAGE 96
ENABLE_AUTOCONNECT 97
ENCRIPT 97, 247
ENCRIPTLEVEL 98, 247
ENMAJORNAME 98
ENQSYSTEMS 100
ERROROWNOTFOUND 100
examples 170
EXT_CP_SUPPT 100
for Adabas data sources 23
for CA IDMS/DB data sources 24
for DB2 for i5/OS data sources 24
for DB2 for Linux, UNIX, and Windows data sources 25
for IMS data sources 26
for Linux, UNIX, and Windows data sources 29
for Microsoft SQL Server data sources 26
for nonrelational data sources 27
for Oracle data sources 27
for the i5/OS operating system 29
for the z/OS operating system 30
for VSAM data sources 28
GDGLOCATE 103, 177
GSBUFSIZE 102
ICUALIAS 103
ICUCNVPROPERTY 103
ICUCONVERTER 105
ICUADIR 107
IMSID 108
JOBCLASS 108
LISTENER 109, 169
LOADCTFILE 110
LOADJOBFILE 111
LOG_CODEPAGE 111
LOG_LINE_LIMIT 112
LOGPATH 112
LOGSID 113
LOWVALUES 113
LRECL 114
MAXTASKS 114
MSGPREFIX 115
MSGPREFIX-HYPHEN 115
MSS_ERRORFILE 116
multiple CAPI connections 166
DBMOVER statements (continued)

USESUP 160
VOLSER 160
VSAM 161
WAITDSN 162
WRT_ERROR_HANDLING 162
DECPNT statement
  DBMOVER configuration file 90
DISABLE_CHAR statement
  DBMOVER configuration file 90
DISABLEDATE statement
  DBMOVER configuration file 91
DES
  encryption 246
determining
  size of alternative log data sets on z/OS 164
Digital Encryption Standard (DES)
  encryption 246
DISABLE_PARTITIONS statement
  DBMOVER configuration file 91
DISP statement
  DBMOVER configuration file 91
DM_RESOURCE statement
  DBMOVER configuration file 92
DM_SUBTASK statement
  DBMOVER configuration file 93
DMX_DIR statement
  DBMOVER configuration file 96
DMXCACHE_DELETEECSA statement
  DBMOVER configuration file 94
DMXCACHE_MAX_MEMORY_MB statement
  DBMOVER configuration file 94
DMXCACHE_MULTIPLEJOBS statement
  DBMOVER configuration file 95
DTLCAPXTIMESTAMP field
  types of reported time stamps by data source 349
DTLADAPREFETCH 340
DTLADAPWD 340
DTLAMCPR DD name 213
DTLAPP 340
DTLCAMAP DD name 213
DTLCONFWRIT 340
DTLCONNVR 340
DTLDATAPWD 341
DTLDB2DEGREE 341
DTLDHFOLDER 341
DTLDHFSIZE 341
DTLDHFSMAP DD name 341
DTLDESCRIBE
  COLUMNS 258, 259
  FKEYS 265
  metadata syntax 252
  Metadata Syntax 252
  PKKEYS 263
  PROCEDURECOLUMNS 256, 257
  PROCEDURES 255, 256
  Qualifiers 268
  RECORDS 261, 262
  SCHEMAS 267
  TABLES 256, 257
DTLMSG_CODEPAGE statement
  DBMOVER configuration file 96
DTLMSGO file
  example 182
  overview 181
DTLMSGO statements
  overview 180

E

ENABLE_AUTOCOMMIT statement
  DBMOVER configuration file 97
ENCRYPT statement
  DBMOVER configuration file 97, 247
equency
  DES 246
  overview 245, 246
  PowerExchange 246
  RC2 246
  setting defaults in the DBMOVER configuration file 247
  triple DES 246
ENCRYPTLEVEL statement
  DBMOVER configuration file 98, 247
ENQMAJORNAME statement
  DBMOVER configuration file 99
ENQSYSTEMS statement
  DBMOVER configuration file 100
ERRORNOTFOUND statement
  DBMOVER configuration file 100
EXT_CP_SUPPT statement
  DBMOVER configuration file 100
extraction maps
  data set-level security 213
  granting access to 215
  RACF class security 213
  resource profiles 214
  security example 216
  security on z/OS 212

F

FIPS 140-2 compliance 232

G

GDGLOCATE statement
  DBMOVER configuration file 101, 177
General tab
  in PowerExchange Data Source wizard 314
GSBUFSIZE statement
  DBMOVER configuration file 102

I

i5/OS
  security 199
i5/OS DBMOVER statements
  DBMOVER configuration file 29
i5/OS security
  resource access requirements 200
ICU-compatible code pages
  adding 282
ICU_ALIAS statement
  DBMOVER configuration file 103
ICUCNVPROPERTY statement
  DBMOVER configuration file 103
ICUCONVERTER statement
  DBMOVER configuration file 105
ICUCDATAIRADIR statement
  DBMOVER configuration file 107
IDMS
  netport jobs 172
  Netport jobs 171
IMS
  Netport jobs 171
  security 220
IMS call considerations
  nonrelational SQL 251
IMS DBMOVER statements
  DBMOVER configuration file 26
IMSID statement
  DBMOVER configuration file 108
infacmd pwx commands
  authorizing users to run (Linux, UNIX, and Windows) 204
  setting up command-handling service for 148
infacmd pwx program
  security for 206
  security for (Linux, UNIX, and Windows) 204
International Components for Unicode (ICU) 272

J
  JOBCLASS statement
  DBMOVER configuration file 108

L
  Linux
    security 204
  Linux, UNIX, and Windows DBMOVER statements
    DBMOVER configuration file 29
LISTENER statement
  DBMOVER configuration file 109, 169
LOADCTFILE statement
  DBMOVER configuration file 110
LOADJOBFILE statement
  DBMOVER configuration file 111
LOG_CODEPAGE statement
  DBMOVER configuration file 111
LOG_LINE_LIMIT statement
  DBMOVER configuration file 112
LOGPATH statement
  DBMOVER configuration file 112
LOGSID statement
  DBMOVER configuration file 113
LOWVALUES statement
  DBMOVER configuration file 113
LRAP CAPI_CONNECTION parameters
  parameters and syntax 51
LRECL statement
  DBMOVER configuration file 114

M
  makeconv utility 283
MAXTASKS statement
  DBMOVER configuration file 114
message destination override 180
metadata syntax
  DTLSDESCRIBE 252
Microsoft SQL Server DBMOVER statements
  DBMOVER configuration file 26
modes of encryption 245
MSGPREFIX statement
  DBMOVER configuration file 115

MSGPREFIX-HYPHEN statement
  DBMOVER configuration file 115
MSQL CAPI_CONNECTION parameters
  parameters and syntax 53
MSS_ERRORFILE statement
  DBMOVER configuration file 116
MVSD2BAF statement
  DBMOVER configuration file 116

N
  NEGSIGN statement
  DBMOVER configuration file 117
  netport jobs
    substitution variables in netport JCL 173
Netport jobs
  configuration 172
IDMS 171
  reasons for using 171
  sample jobs 172
NETPORT statement
  DBMOVER configuration file 117
NODE statement
  DBMOVER configuration file 120
NOGETHOSTBYNAME statement
  DBMOVER configuration file 121
nonrelational DBMOVER statements
  DBMOVER configuration file 27
  nonrelational SQL
    IMS call considerations 251
    overview 248
    supported syntax 248
NRDDB_WRITE_CHAR_NULL_FILL statement
  DBMOVER configuration file 122
NRDDB_WRITE_NUM_NULL_FILL statement
  DBMOVER configuration file 122
NUMERICSIGN statement
  DBMOVER configuration file 123

O
  ODBASUPP statement
  DBMOVER configuration file 123
ODBC
  CAPXRT Properties tab in PowerExchange Data Source wizard 320
  DB2/S390 Bulk Load Properties tab in PowerExchange Data Source wizard 322
  General tab in PowerExchange Data Source wizard 314
  PowerExchange Data Source tab 312
  S/400 Properties tab in PowerExchange Data Source wizard 317
ODBC data source
  creating (UNIX) 325
  creating on Windows 310
ODBC driver
  PowerExchange 311
ODBC SQL C datatypes
  SQL_C_BINARY 347
  SQL_C_BIT 347
  SQL_C_CHAR 347
  SQL_C_DATE 347
  SQL_C_DOUBLE 347
  SQL_C_FLOAT 347
  SQL_C_LONG 347
  SQL_C_SHORT 347
  SQL_C_SSHORT 347
  SQL_C_SLONG 347

374  Index
ODBC SQL C datatypes (continued)
SQL_C_STINYINT 347
SQL_C_TIME 347
SQL_C_TIMESTAMP 347
SQL_C_TINYINT 347
SQL_C_ULONG 347
SQL_C_USHORT 347
SQL_C_UTINYINT 347

ODBC SQL datatypes
SQL_BINARY 347
SQL_BIT 347
SQL_CHAR 347
SQL_DATE 347
SQL_DECIMAL 347
SQL_DOUBLE 347
SQL_INTEGER 347
SQL_REAL 347
SQL_SMALLINT 347
SQL_TIME 347
SQL_TIMESTAMP 347
SQL_TINYINT 347
SQL_VARBINARY 347
SQL_VARCHAR 347

OpenSSL 240
operating system authentication
for PowerExchange 135, 195
operating system-specific DBMOVER statements
DBMOVER configuration file 28
ORA_ERRORFILE statement
DBMOVER configuration file 124
Oracle DBMOVER statements
DBMOVER configuration file 27
ORACLE_CAPTURE_TYPE statement
DBMOVER configuration file 124
ORACLE_UNHANDLED_NUMASCHAR statement
DBMOVER configuration file 125
ORACLECODEPAGE
and globalization 280
ORACLECODEPAGE statement
DBMOVER configuration file 126
ORACLEID statement
DBMOVER configuration file 127
ORAD CAPI_CONNECTION parameters
parameters and syntax 96
ORACL CAPI_CONNECTION parameters
parameters and syntax 99
order of precedence
CAPI statements 168
OUSP statement
DBMOVER configuration file 129
overview
PowerExchange 1
PowerExchange Agent 1
PowerExchange Condense 1
PowerExchange ECCR 1
PowerExchange Listener 1
PowerExchange Logger for Linux, UNIX, and Windows 1
PowerExchange Logger for MVS 1
PowerExchange Navigator 1

P

PC_AUTH statement
DBMOVER configuration file 130
personal certificate, creating 241
PIPE statement
DBMOVER configuration file 130

POLLTIME statement
DBMOVER configuration file 130
PowerExchange
components 1
configuring operating system authentication 135, 195
encryption 246
ODBC driver 311
overview 1
PowerExchange Agent
overview 1
PowerExchange alternative log files 164
PowerExchange Condense
overview 1
PowerExchange Data Source tab
in PowerExchange Data Source wizard 312
PowerExchange Data Source wizard
AS/400 Properties tab 317
CAPX Properties tab 319
CAPXRT Properties tab 320
DB2/S390 Properties tab 322
General tab 314
overview 311
PowerExchange Data Source tab 312
PowerExchange datatypes
DTLNET_BIN 347
DTLNET_BIT 347
DTLNET_CHAR 347
DTLNET_DATE 347
DTLNET_DOUBLE 347
DTLNET_FLOAT 347
DTLNET_MONEY 347
DTLNET_NUM16 347
DTLNET_NUM16U 347
DTLNET_NUM32 347
DTLNET_NUM32U 347
DTLNET_NUM64 347
DTLNET_NUM64U 347
DTLNET_NUM8 347
DTLNET_NUM8U 347
DTLNET_NUMCHAR 347
DTLNET_PACKED 347
DTLNET_STRING 347
DTLNET_TIME 347
DTLNET_TIMESTAMP 347
DTLNET_UPACKED 347
DTLNET_UZONED 347
DTLNET_VARBINARY 347
DTLNET_VARCHAR 347
DTLNET_ZONED 347
PowerExchange ECCR
overview 1
PowerExchange jobs
data set access requirements 208
PowerExchange Listener
overview 1
PowerExchange Listener commands
authorizing users to run (i5/OS) 202
authorizing users to run (Windows) 204
security for (Linux, UNIX, and Windows) 204
PowerExchange Logger for Linux, UNIX, and Windows
overview 1
PowerExchange Logger for MVS
overview 1
PowerExchange message log
options 163
PowerExchange message log file 163
PowerExchange Navigator
overview 1
SQL escape sequences (continued)
DTLNoupdatececep 343
Dtloracoll 343
Dtloracomm 343
Dtlorainst 343
Dtloraschema 343
Dtlvorpdf 343
Dtlrejectfile 344
Dtlsesid 344
Dtlstopenerrors 344
Dtlstriporderby 344
Dtltimeout 344
Dtludbdb 344
Dtlupdeleins 344
Dtlxtraschema 345
Dtxtype 345
ODBC 339
with PowerCenter 339
SSL
architecture, PowerExchange 232
security on PowerExchange network 231
support 231
SSL statement
DBMOVER configuration file 140
SSL_ALLOW_SELFIGNED statement
DBMOVER configuration file 141
SSL_CIPHER_LIST statement
DBMOVER configuration file 141
SSL_CONTEXT_METHOD statement
DBMOVER configuration file 142
SSL_REQ_CLNT_CERT statement
DBMOVER configuration file 143
SSL_REQ_SRVR_CERT statement
DBMOVER configuration file 143
statistics logging
using SMF 183
STATS statement
DBMOVER configuration file 144
SUBMITTIMEOUT statement
DBMOVER configuration file 145
summary
DBMOVER statements 4
SUP_FUNC statement
DBMOVER configuration file 146
SUP_REQUEST statement
DBMOVER configuration file 146
SUP_SNAME statement
DBMOVER configuration file 146
SUP_SSTYPE statement
DBMOVER configuration file 147
SUPPRESS_DATA_LOGGING statement
DBMOVER configuration file 147
SVNODE statement
DBMOVER configuration file 148, 169
syntax
PowerExchange sign-on file 225
syntax rules
DBMOVER statements 33
SYNOUT_TIMESTAMP statement
DBMOVER configuration file 149

TCPIP_DIAGNOSTICS_TRACE statement
DBMOVER configuration file 149
TCPIP_SHOW_POOLING statement
DBMOVER configuration file 150
TCPIPBUF_SIZE statement
DBMOVER configuration file 151
TCPIPVER statement
DBMOVER configuration file 151
TEMPHLQ statement
DBMOVER configuration file 151
TEXT_EOF_FOR_BINARY statement
DBMOVER configuration file 152
TIMEZONE statement
DBMOVER configuration file 152
TRACE statement
DBMOVER configuration file 153
TRACING statement
DBMOVER configuration file 153, 163, 164
triple DES
encryption 246

U
UDB CAPI_CONNECTION parameters
parameters and syntax 64
UNIT statement
DBMOVER configuration file 159
UNIX
security 204
UOWC CAPI_CONNECTION parameters
parameters and syntax 68
USE_TYPE1_FALLBACKS statement
DBMOVER configuration file 159
user credentials
authenticating (Linux and UNIX) 205
authenticating (Windows) 205
USER statement
sign-on file 226, 227
USESUP statement
DBMOVER configuration file 160

V
VOLSER statement
DBMOVER configuration file 160
VSAM DBMOVER statements
DBMOVER configuration file 28
VSAM statement
DBMOVER configuration file 161

W
WAITDSN statement
DBMOVER configuration file 162
Windows
security 204
WRT_ERROR_HANDLER statement
DBMOVER configuration file 162

Z
z/OS
security 206