



Informatica® PowerExchange
10.2

PowerExchange Interfaces for PowerCenter

Informatica PowerExchange PowerExchange Interfaces for PowerCenter

10.2

September 2017

© Copyright Informatica LLC 2006, 2018

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

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

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

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

The information in this documentation is subject to change without notice. If you find any problems in this documentation, please report them to us in writing at Informatica LLC 2100 Seaport Blvd. Redwood City, CA 94063.

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

Publication Date: 2018-06-08

Table of Contents

Preface	9
Informatica Resources.	9
Informatica Network.	9
Informatica Knowledge Base.	10
Informatica Documentation.	10
Informatica Product Availability Matrixes.	10
Informatica Velocity.	10
Informatica Marketplace.	10
Informatica Global Customer Support.	11
 Part I: Introduction.....	 12
 Chapter 1: PowerExchange Interfaces for PowerCenter.....	 13
PowerExchange Interfaces for PowerCenter Overview.	13
PowerExchange Client for PowerCenter (PWXPC).	14
Bulk Extraction Mode.	16
CDC Batch and Continuous Extraction Modes.	16
CDC Real-time Extraction Mode.	17
PowerExchange ODBC Drivers.	18
 Part II: PowerExchange Client for PowerCenter (PWXPC).....	 20
 Chapter 2: Installing PowerExchange Client for PowerCenter (PWXPC).....	 21
PWXPC Installation Overview.	21
Installing and Configuring PWXPC.	21
Establishing Restart Points for CDC Sessions.	22
Modifying the PowerExchange Configuration Files.	22
PowerExchange Interoperability with PowerCenter.	23
 Chapter 3: Working with Mappings.....	 24
Working with Mappings Overview.	24
Source and Target Definitions.	25
Working with Relational Source and Target Definitions.	25
Working with DB2 Definitions.	25
Working with Microsoft SQL Server Definitions.	28
Working with Oracle Definitions.	30
Editing Relational Source and Target Definitions.	32
Working with Nonrelational Source and Target Definitions.	32
Importing Nonrelational Source Definitions.	33
Importing Nonrelational Target Definitions.	35

Viewing Nonrelational Source and Target Definition Details.	37
Editing Nonrelational Source and Target Definitions	39
Working with Extraction Map Definitions.	40
Viewing Extraction Map Definition Details	42
Editing an Extraction Map Definition	44
Previewing PowerExchange Data in Designer.	44
Previewing PowerExchange Relational Source or Target Data.	44
Previewing PowerExchange Nonrelational Source or Target Data.	45
Previewing PowerExchange Change Data.	46
Group Source Processing in PowerExchange.	47
Group Source Processing in Bulk Data Movement Sessions with Nonrelational Data Sources.	47
Group Source Processing in CDC Sessions.	49
Multiple-Record Writes to Nonrelational Targets.	50
Group Processing with Sequencing.	50
Sequencing Support in Source and Target Definitions.	51
Source Qualifier Transformations.	52
Source Qualifier Transformation Datatypes.	52
Creating Source Qualifiers.	53
Editing Source Qualifier Transformations.	53
Lookup Transformations.	53
Usage Considerations for Lookup Transformations.	53
Limitations on Nonrelational Uncached Lookups.	54
Configuring Lookups for IMS.	55
Configuring Lookups for CDC Data.	55
Stored Procedure Transformations.	56
Usage Considerations for Stored Procedure Transformations.	56
Implementing a Stored Procedure Transformation in a Mapping.	57
Chapter 4: Connections.	58
Connections Overview.	58
Connection Types for Extracting Source Data.	59
Connection Types for Loading Target Data.	60
Configuring PowerExchange Connections.	61
DB2 Relational Connections.	62
DB2 CDC Application Connections.	70
NRDB Batch Application Connections.	77
NRDB CDC Application Connections.	82
NRDB Lookup Relational Connections.	88
Microsoft SQL Server Relational Connections.	91
Microsoft SQL Server CDC Application Connections.	94
Oracle Relational Connections.	99
Oracle CDC Application Connections.	102
Configuring Connection Attributes.	108

Common Connection Attributes.	109
Batch Application and Relational Connection Attributes.	112
CDC Application Connection Attributes.	113
Connection Attributes for CDC Commit Processing	122
Examples of Using CDC Commit Processing Attributes.	124
Connection Attributes for Multiple-Record Writes.	126
Asynchronous Network Communication.	127
Asynchronous Read and Write Processing.	127
Timeout Processing.	127
Configuring Connection and Network Operation Timeouts.	128
Chapter 5: Working with Sessions.	129
Working with Sessions Overview.	129
Extracting Data in Batch Mode.	129
Data Extraction from Multiple Sequential or VSAM ESDS Files.	130
Overrides for the IMS Access Method and Related Properties.	130
Configuring Batch Mode Sessions for Nonrelational Sources.	131
Configuring Session Properties for Relational Batch Mode Sessions.	135
Extracting CDC Data in Change and Real-time Modes.	137
PowerExchange Extraction Maps.	137
Data Extraction from Multiple CDC Sources in a Session.	138
Session Properties for Change and Real-time Mode Sessions.	138
Asterisk Wildcard in i5/OS Library or File Overrides.	142
Treating Source Rows as Updates.	142
Loading Data to PowerExchange Targets.	143
Configuring Sessions to Load Data to Relational Targets.	143
Configuring Sessions to Load Data to Nonrelational Targets.	144
Considerations for Sessions that Perform Multiple-Record Writes.	149
Sequencing and Queuing Cache for Multiple-Record Writes.	149
Row Statistics for Multiple-Record Writes.	151
Multiple-Record Write Session Properties for Sources.	151
Multiple-Record Write Session Properties for Targets.	152
Constraint-Based Loading for Relational Targets.	153
Configuring Constraint-Based Loading for Relational Targets.	153
Variables for Session and Connection Properties.	154
Workflows that Connect to a PowerExchange Listener Service.	155
Configuring PowerExchange Services for Workflows.	156
Pipeline Partitioning.	156
Commit Processing for Bulk Data Movement Sessions.	156
Commit Processing for CDC Sessions.	157
Stored Procedure Calls in Pre- and Post-Session SQL Commands.	157
Preserving Low Values in Source Character Fields.	158

Chapter 6: Restart and Recovery.	159
Restart and Recovery Overview.	159
Understanding PWXPC Restart and Recovery.	160
Session Recovery.	160
Recovery Tables.	160
Recovery State Table.	161
Recovery State File.	162
Restart Token File.	162
How Restart Points Are Determined.	163
Initializing and Running CDC Sessions.	166
Ending CDC Sessions.	167
Creating Recovery Tables.	167
Recovery Table Creation with PowerExchange Targets.	167
Creating the Recovery Tables Manually.	168
Configuring the Restart Token File.	169
Restart Token File Control Statement Syntax.	169
Restart Token File Control Statement Parameters.	169
Restart Token File Example.	172
Using PWXPC Restart and Recovery.	172
Enabling Recovery for a CDC Session.	173
Configuring CDC Sessions for Restart Processing.	174
Application Name Requirement for Restart Processing.	175
Generating Current Restart Tokens for CDC Sessions.	175
Starting CDC Sessions.	176
Stopping CDC Sessions.	178
Changing CDC Sessions and Restarting from the Point of Interruption.	178
Recovering from CDC Session Failures.	180
Managing Session Log and Restart Token File History.	181
 Chapter 7: Flexible Target Key Transformation.	 182
Flexible Target Key Transformation Overview.	182
Configuring PowerExchange Extraction Maps.	183
Configuring Mappings with Flexible Target Key Transformations.	184
Configuring Flexible Target Key Transformations with Multiple Sources.	185
 Part III: PowerExchange ODBC.....	 187
 Chapter 8: Installing PowerExchange ODBC.	 188
Installing PowerExchange ODBC Overview.	188
Modifying the PowerExchange Configuration Files.	188
Creating ODBC Data Sources.	189
Creating an ODBC Data Source on Windows.	189

Creating a PowerExchange ODBC Data Source on Linux or UNIX.	190
Chapter 9: Working with Mappings for PowerExchange ODBC.	191
Working with Mappings for PowerExchange ODBC Overview.	191
Working with Source and Target Definitions for PowerExchange Batch (ODBC).	192
Importing a Source or Target Definition for PowerExchange ODBC.	192
Working with Source Definitions for PowerExchange Change or Real-time (ODBC).	193
Importing a Change Data Capture Source Definition for PowerExchange ODBC.	193
Chapter 10: Configuring Connections for PowerExchange ODBC.	195
Configuring Connections for PowerExchange ODBC Overview.	195
Working with Connections for PowerExchange ODBC.	195
Connection Types for Extracting Source Data for PowerExchange ODBC.	195
Connection Types for Loading Target Data for PowerExchange ODBC.	196
Configuring Attributes for Connections for PowerExchange ODBC.	196
Chapter 11: Working with Sessions for PowerExchange ODBC.	198
Working with Sessions for PowerExchange ODBC Overview.	198
Pipeline Partitioning for PowerExchange ODBC.	198
Extracting Data from PowerExchange in Batch Mode (ODBC).	199
Configuring Properties for Nonrelational Batch Mode Sessions (ODBC).	199
Configuring Properties for Relational Batch Mode Sessions (ODBC).	200
Extracting Data from PowerExchange in Change and Real-time Mode (ODBC).	200
Configuring Properties for Change and Real-time Mode Sessions (ODBC).	200
Configuring a Session to Load Data to PowerExchange Targets (ODBC).	201
Chapter 12: PowerExchange Restart and Recovery.	202
PowerExchange Restart and Recovery Overview.	202
Restart and Recovery with PowerExchange ODBC.	202
Managing Restart Tokens with PowerExchange ODBC.	203
Appendix A: PowerExchange Interfaces for PowerCenter Tips.	204
Organizing Sources by Map Type in the Designer.	204
Filtering Source Data with PWXPC.	205
Filter Override Examples.	206
DTLREXE to Submit MVS Batch Jobs.	207
Using Pre-SQL and Post-SQL Statements to Manage Files and Tables.	208
Creating Empty Files.	208
Truncating a Target Table.	212
Inserting Rows into a Table.	213
Appendix B: Datatypes and Code Pages.	214
Datatypes and Code Pages Overview.	214

PowerExchange Nonrelational Datatypes and Transformation Datatypes.	214
Restrictions on Relational Datatypes.	216
LOB Datatypes.	217
DB2 for z/OS TIMESTAMP Datatype.	217
Reading and Writing Binary Data in PowerExchange Client for PowerCenter	218
Code Pages.	218
Appendix C: PowerExchange Interfaces for PowerCenter Troubleshooting..	219
Troubleshooting for PowerExchange Interfaces for PowerCenter.	219
Index.	221

Preface

The *PowerExchange Interfaces for PowerCenter* publication describes the Informatica® PowerExchange® Client for PowerCenter (PWXPC) and ODBC interfaces between PowerExchange and PowerCenter®. This publication is intended for developers and administrators who are responsible for creating, running, and administering workflows and sessions that interface with PowerExchange.

This publication assumes that you have knowledge of relational database concepts and the database engines and nonrelational files in your environment. This guide also assumes that you are familiar with the basic operation of PowerExchange and PowerCenter.

This publication supplements the information in the PowerCenter library, including the *PowerCenter Designer Guide*, *PowerCenter Transformation Guide*, *PowerCenter Workflow Basics Guide*, *PowerCenter Advanced Workflow Guide*, and *Informatica Administrator Guide*.

This publication applies to the following PowerExchange products, including PowerExchange change data capture (CDC) where applicable:

- PowerExchange for Adabas
- PowerExchange for CA Datacom
- PowerExchange for CA IDMS
- PowerExchange for DB2® for i5/OS
- PowerExchange for DB2 for Linux, UNIX, and Windows
- PowerExchange for DB2 for z/OS
- PowerExchange for IMS
- PowerExchange for Flat Files on Linux, UNIX, and Windows
- PowerExchange for Oracle
- PowerExchange for SQL Server
- PowerExchange for VSAM

Informatica Resources

Informatica Network

Informatica Network hosts Informatica Global Customer Support, the Informatica Knowledge Base, and other product resources. To access Informatica Network, visit <https://network.informatica.com>.

As a member, you can:

- Access all of your Informatica resources in one place.
- Search the Knowledge Base for product resources, including documentation, FAQs, and best practices.
- View product availability information.
- Review your support cases.
- Find your local Informatica User Group Network and collaborate with your peers.

Informatica Knowledge Base

Use the Informatica Knowledge Base to search Informatica Network for product resources such as documentation, how-to articles, best practices, and PAMs.

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

Informatica Documentation

To get the latest documentation for your product, browse the Informatica Knowledge Base at https://kb.informatica.com/_layouts/ProductDocumentation/Page/ProductDocumentSearch.aspx.

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

Informatica Product Availability Matrixes

Product Availability Matrixes (PAMs) indicate the versions of operating systems, databases, and other types of data sources and targets that a product release supports. If you are an Informatica Network member, you can access PAMs at

<https://network.informatica.com/community/informatica-network/product-availability-matrixes>.

Informatica Velocity

Informatica Velocity is a collection of tips and best practices developed by Informatica Professional Services. 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 are an Informatica Network member, you can access Informatica Velocity resources at <http://velocity.informatica.com>.

If you have questions, comments, or ideas about Informatica Velocity, contact Informatica Professional Services at ips@informatica.com.

Informatica Marketplace

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

Informatica Global Customer Support

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

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

<http://www.informatica.com/us/services-and-training/support-services/global-support-centers>.

If you are an Informatica Network member, you can use Online Support at <http://network.informatica.com>.

Part I: Introduction

This part contains the following chapter:

- [PowerExchange Interfaces for PowerCenter, 13](#)

CHAPTER 1

PowerExchange Interfaces for PowerCenter

This chapter includes the following topics:

- [PowerExchange Interfaces for PowerCenter Overview, 13](#)
- [PowerExchange Client for PowerCenter \(PWXPC\), 14](#)
- [PowerExchange ODBC Drivers, 18](#)

PowerExchange Interfaces for PowerCenter Overview

You can use the following interfaces to extract and load data through PowerExchange when you use PowerCenter:

- **PowerExchange Client for PowerCenter (PWXPC).** Part I describes PWXPC. You can use it to extract and load data through PowerExchange for a variety of datatypes on a variety of platforms. PWXPC is fully integrated into PowerCenter.
- **PowerExchange ODBC.** Part II describes the PowerExchange ODBC interface. You can use PowerExchange ODBC connections with PowerCenter to extract and load data through PowerExchange for a variety of datatypes on a variety of platforms.

Note: When connecting to PowerExchange, Informatica recommends that you use PWXPC instead of PowerExchange ODBC. PWXPC has additional functionality as well as improved performance and superior CDC recovery and restart.

The following table compares the interface functionality of the PowerExchange Client for PowerCenter and PowerExchange ODBC:

PWXPC	ODBC	Function	Description
Yes	Yes	Extracts bulk data and changed data	PowerExchange extracts bulk data for relational and nonrelational sources and changed data for CDC sources.
Yes	No	Extracts data for multiple sources or records in a single pass	PowerExchange uses group source to extract changed data for multiple data sources or bulk data for multiple record types in a VSAM and sequential files in a single pass.

PWXPC	ODBC	Function	Description
Yes	No	Saves target data and CDC restart information in a single commit	CDC restart information is stored in the same database as the relational target table or MQ queue. The restart information is updated in the same commit as the target data providing guaranteed restart and recovery for CDC data.
Yes	No	Uses PowerCenter graceful stop for real-time sessions	PowerCenter stops real-time sessions after all data in the pipeline is written to the targets.
Yes	No	Uses the change indicator to determine the type of change record	Each change record indicates whether it is an insert, update, or delete. When the change indicator is used, an Update Strategy transformation is not required to process inserts, updates, and deletes.
Yes	No	Creates source definitions from PowerExchange extraction maps	Extraction maps contain the PowerExchange auto-generated columns minimizing modification of the source definition in Designer.
Yes	No	Uses the file name in the PowerCenter source definition	PowerCenter source definition can specify the file name and override the file name specified in the PowerExchange data map.

PowerExchange Client for PowerCenter (PWXPC)

PowerExchange Client for PowerCenter is installed with PowerCenter and integrates PowerExchange and PowerCenter to extract relational, nonrelational, and changed data. PWXPC also loads relational and nonrelational data in batch mode.

Use the following modes to extract relational and nonrelational data:

- Bulk data movement
- Change data capture (CDC) real time
- CDC batch extraction mode from condense files
- CDC continuous extraction mode from condense files

The following table lists the database types that PowerExchange Client for PowerCenter can access to extract data or to load data:

Database Type	Bulk Mode Extract/Load	CDC Real-time Extraction Mode	CDC Batch Extraction Mode	CDC Continuous Extraction Mode
Adabas	Yes/Yes	Yes	Yes	No
Datacom	Yes/No	Yes	Yes	No
DB2 for z/OS	Yes/Yes	Yes	Yes	No
DB2 for i5/OS	Yes/Yes	Yes	Yes	No

Database Type	Bulk Mode Extract/Load	CDC Real-time Extraction Mode	CDC Batch Extraction Mode	CDC Continuous Extraction Mode
DB2 for Linux, UNIX, and Windows	Yes/Yes	Yes	Yes	Yes
IDMS	Yes/No	Yes	Yes	No
IMS	Yes/Yes	Yes	Yes	No
MSSQL	Yes/Yes	Yes	Yes	Yes
Oracle	Yes/Yes	Yes	Yes	Yes
Sequential files	Yes/Yes	No	No	No
VSAM Note: For VSAM ESDS and RRDS data sets, only inserts are allowed. For VSAM KSDS data sets, inserts, updates, and deletes are allowed.	Yes/Yes	Yes	Yes	No

PowerExchange group source reads data for data sets and files with multiple record types or multiple CDC data sources in a single pass. PWXPC uses PowerExchange group source to extract changed data from the change stream. PWXPC also uses group source to extract data for multiple record types from VSAM data sets, sequential files, and IMS unload data sets. As a result, PWXPC connections can process data faster than PowerExchange ODBC connections and reduce PowerExchange resource consumption on the source or extraction platform.

The following table lists the PowerExchange database types that read sources in a single pass during extraction:

Database Type	Bulk Extraction Mode	CDC Real-time Extraction Mode	CDC Batch Extraction Mode	CDC Continuous Extraction Mode
Adabas	No	Yes	Yes	No
Datacom	No	Yes	Yes	No
DB2 for z/OS	No	Yes	Yes	No
DB2 for i5/OS	No	Yes	Yes	No
DB2 for Linux, UNIX, and Windows	No	Yes	Yes	Yes
IDMS	No	Yes	Yes	No
IMS	Yes Note: IMS unload data sets only.	Yes	Yes	No
Microsoft SQL Server	No	Yes	Yes	Yes

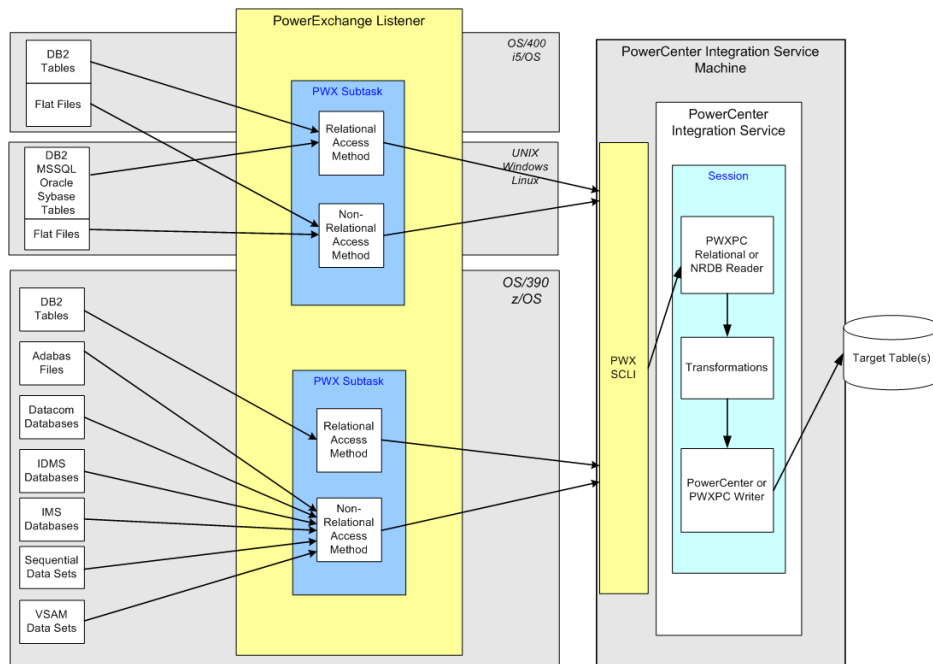
Database Type	Bulk Extraction Mode	CDC Real-time Extraction Mode	CDC Batch Extraction Mode	CDC Continuous Extraction Mode
Oracle	No	Yes	Yes	Yes
Sequential files	Yes	No	No	No
VSAM	Yes	Yes	Yes	No

Bulk Extraction Mode

Use PWX batch application and relational connections to extract and load data for relational databases and nonrelational data sets and files through PowerExchange. PWXPC connects to PowerExchange through the PowerExchange Call Level Interface (SCLI).

When you use PWXPC, you can extract all records for multiple record types in a single pass for IMS unload data sets, VSAM data sets, and sequential files. In contrast, PowerExchange ODBC connections read a single record type at a time, which requires multiple passes of the data.

The following figure shows the data flow of source data from PowerExchange through PWXPC and PowerCenter to the target tables:



CDC Batch and Continuous Extraction Modes

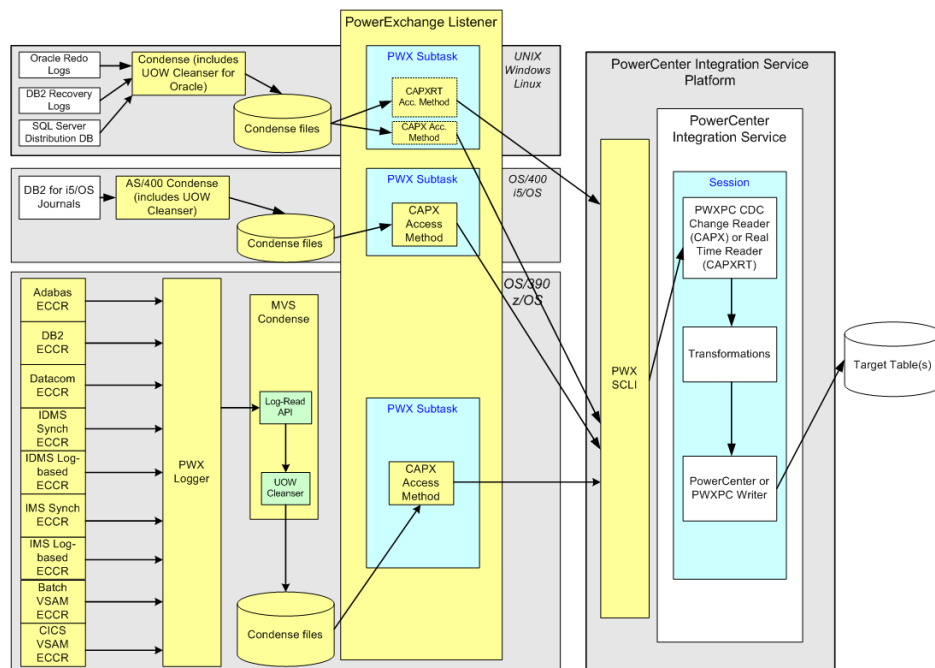
PowerExchange Condense captures changes for relational tables and nonrelational data sets and files from the change stream and writes the changed data to condense files. The change stream contains the source data changes in time sequence order. As a result, the changes in a single unit of work are not contiguous in the change stream. The UOW Cleanser reconstructs the units of work into complete and consecutive units of work based on transaction end time. PowerExchange Condense writes the reconstructed units of work to condense files after which you can use extract the changed data.

PWXPC extracts changed data from PowerExchange condense files in two different modes:

- **Batch extraction mode from condense files.** Use PWX CDC Change connections to extract changed data from condense files in batch extraction mode. PWXPC uses the PowerExchange CAPX access method when processing data with CDC Change connections. PowerExchange stops the extraction after the data from all condense files is read.
- **Continuous extraction mode from condense files.** Use PWX CDC Real Time connections for Linux, UNIX, and Windows sources to extract changed data from condense files in continuous extraction mode. PWXPC uses the PowerExchange CAPXRT access method when processing data with CDC Real Time connections. PowerExchange runs the extraction until stopped. For more information, see the *PowerExchange CDC Guide for Linux, UNIX, and Windows*.

PWXPC connects to PowerExchange through the PowerExchange Call Level Interface (SCLI). PowerExchange reads the changed data from each condense file once for all sources in the mapping in a single pass.

The following figure shows the data flow of condensed changed data from PowerExchange through PWXPC and PowerCenter to the target tables:



CDC Real-time Extraction Mode

You can extract changed data for relational tables and nonrelational data sets and files directly from the change stream in real time. The change stream contains the source data changes in time sequence order. As a result, the changes in a single unit of work are not contiguous in the change stream. The UOW Cleanser reconstructs the units of work into complete and consecutive units of work based on transaction end time. The UOW Cleanser is incorporated into UDB and SQL Server CDC but is a separate task for all other PowerExchange CDC sources.

Use PWX CDC Real Time application connections to extract changed data for relational tables or nonrelational data sets and files directly from the change stream. PWXPC connects to PowerExchange through the PowerExchange Call Level Interface (SCLI) and specifying the PowerExchange CAPXRT access method. PowerExchange reads the changed data from the change stream for all sources in the mapping in a single pass. PWXPC real-time sessions run for a specified period or continuously until stopped.

The following table lists the PowerExchange ODBC extraction and load modes by source type:

Database Type	Batch Mode Extract	Batch Mode Load	CDC Batch Extraction Mode	CDC Real-time Extraction Mode
Adabas	Yes	Yes	Yes	Yes
Datacom	Yes	No	Yes	Yes
DB2 for z/OS	Yes	Yes	Yes	Yes
DB2 for i5/OS	Yes	Yes	Yes	Yes
DB2 for Linux, UNIX, and Windows	Yes	Yes	Yes	Yes
IDMS	Yes	No	Yes	Yes
IMS	Yes	Yes	Yes	Yes
MSSQL	No	Yes	Yes	Yes
Oracle	No	Yes	Yes	Yes
Sequential/flat files	Yes	Yes	No	No
VSAM Note: For VSAM ESDS and RRDS data sets, only inserts are allowed. For VSAM KSDS data sets, inserts, updates, and deletes are allowed.	Yes	Yes	Yes	Yes

Part II: PowerExchange Client for PowerCenter (PWXPC)

This part contains the following chapters:

- [Installing PowerExchange Client for PowerCenter \(PWXPC\), 21](#)
- [Working with Mappings, 24](#)
- [Connections, 58](#)
- [Working with Sessions, 129](#)
- [Restart and Recovery, 159](#)
- [Flexible Target Key Transformation, 182](#)

CHAPTER 2

Installing PowerExchange Client for PowerCenter (PWXPC)

This chapter includes the following topics:

- [PWXPC Installation Overview, 21](#)
- [Installing and Configuring PWXPC, 21](#)
- [PowerExchange Interoperability with PowerCenter, 23](#)

PWXPC Installation Overview

The PowerExchange Client for PowerCenter (PWXPC) is a native PowerCenter plug-in that is installed with PowerCenter.

To use PWXPC, you must also install PowerExchange on the PowerCenter Client and PowerCenter Integration Service machines.

Note: If the appropriate version of PowerExchange is not installed and available on the PowerCenter Client platform, the **Import from PowerExchange** dialog box will not function.

Installing and Configuring PWXPC

To install PWXPC, complete the PowerCenter installation steps for the PowerCenter Integration Service and PowerCenter Client.

Also, install and configure PowerExchange on the machines where the PowerCenter Integration Service and PowerCenter Client run.

To use PWXPC, perform the following PowerExchange configuration tasks:

- If you use PowerExchange Change Data Capture (CDC), configure the restart token file to establish an extraction restart point for CDC sessions.
- To configure PWXPC on the PowerCenter Integration Service and PowerCenter Client, edit PowerExchange dbmover.cfg configuration files on those PowerCenter nodes.

In these dbmover.cfg files, add NODE statements that point to the PowerExchange Listeners to which you want PWXPC to connect.

Establishing Restart Points for CDC Sessions

If you use PowerExchange Change Data Capture (CDC), after you upgrade PowerCenter, you must configure the restart token file to establish an extraction restart point for CDC sessions.

If you upgrade from PowerCenter 8.6, you can warm start CDC sessions after the PowerCenter upgrade.

If you upgrade from a PowerCenter release earlier than 8.6 and use enhanced restart for CDC sessions, you must cold start all CDC sessions.

Modifying the PowerExchange Configuration Files

You must define nodes for the PowerExchange Listeners in the PowerExchange configuration file, `dbmover.cfg`, on the PowerCenter Integration Service and Client machines. The `NODE` statements specify the information that PowerExchange uses to communicate with Listeners that access the source or target databases or files.

Note: You can also use PowerExchange local mode if the data resides on the same machine as the PowerCenter Integration Service. In local mode, a PowerExchange Listener is not required. If you use local mode, specify **local** in the Location property in PWXPC connections. You do not need to update the PowerExchange `dbmover.cfg` file.

Local mode is not available on 32-bit Windows systems.

Adding PowerExchange Nodes on the PowerCenter Client Machine

PWXPC connects to a PowerExchange Listener to import source and target definitions. PWXPC retrieves the metadata from the machine indicated in the `NODE` statement or retrieves the metadata locally in local mode.

Update the `dbmover.cfg` file on the PowerCenter Client machine to provide the appropriate `NODE` statements for the PowerExchange Listeners to which you need to connect.

The nodes you add to the `dbmover.cfg` file are specified in the Location field in Import from PowerExchange dialog box.

Adding PowerExchange Nodes on the PowerCenter Integration Service Machine

PWXPC connects to the appropriate PowerExchange Listener to extract data from a source or load data to a target.

Update the `dbmover.cfg` file on the PowerCenter Integration Service machine to provide the appropriate `NODE` statements for the PowerExchange Listeners to which you need to connect.

The nodes you add to the `dbmover.cfg` file on the PowerCenter Integration Service machine are specified in the Location value of the PWXPC connections.

Adding PowerExchange Nodes in the `dbmover.cfg` File

Use the following procedure to add PowerExchange nodes in the `dbmover.cfg` file:

1. Locate the `dbmover.cfg` file in the PowerExchange root directory.
2. Open the file with a text editor.
3. Create a node for each PowerExchange Listener that you want to register. Use the following format:

```
NODE=(node_name,TCPIP,hostname_or_ipaddress,port_number)
```

The *node name* is a logical name used to reference the PowerExchange Listener. The *hostname_or_ipaddress* and *port_number* are the host name or IP address and the port number of the PowerExchange Listener.

4. Save your changes.

The following example shows several nodes in a dbmover.cfg file:

```

/*****
/* PowerExchange Configuration File
*****/
NODE=(AS400_DB2,TCPIP,AS400_1,2480)
NODE=(MVS1_VSAM,TCPIP,MVS1,5539)
NODE=(MVS2_DB2,TCPIP,10.3.4.5,5538)

```

PowerExchange Interoperability with PowerCenter

PowerCenter includes the PowerExchange Client for PowerCenter (PWXPC). To use PWXPC to connect to PowerExchange, you must verify that the PowerCenter and PowerExchange versions are compatible.

The following table describes the supported combinations of PowerExchange and PowerCenter versions:

PWX ¹ Version	PWC ² 9.6.0	PWC 9.6.1	PWC 10.0	PWC 10.1	PWC 10.1.1	PWC 10.2
9.6.0	Yes	-	-	-	-	-
9.6.1	Yes	Yes	-	-	-	-
10.0	-	-	Yes	-	-	-
10.1	-	-	Yes	Yes	-	-
10.1.1	-	-	Yes	Yes	Yes	-
10.2	-	-	Yes	Yes	Yes	Yes
1. PWX stands for PowerExchange. 2. PWC stands for PowerCenter.						

PowerExchange hotfixes are certified with the latest available PowerCenter hotfix.

Restrictions:

- If you run the PowerExchange Listener Service or PowerExchange Logger Service in the Informatica domain, the Informatica domain must be at the same version, release, and modification (v.r.m) level as PowerExchange.

To run different versions of the Informatica domain and PowerExchange, you must start the PowerExchange Listener or PowerExchange Logger for Linux, UNIX, and Windows by running the dtllst or pwxcl command-line program, respectively.

- If you are using PowerCenter 10.0 with PowerExchange 10.1, local mode is not supported for bulk data movement sessions. That is, do not specify **local** in the Location property in PWXPC connections for bulk data movement sessions. Instead, configure the PowerCenter Integration Server to connect to the PowerExchange Listener on the machine. To do so, define a NODE statement for the machine in the DBMOVER configuration file, and enter this node name in the Location property for the connection.

CHAPTER 3

Working with Mappings

This chapter includes the following topics:

- [Working with Mappings Overview, 24](#)
- [Source and Target Definitions, 25](#)
- [Working with Relational Source and Target Definitions, 25](#)
- [Working with Nonrelational Source and Target Definitions, 32](#)
- [Working with Extraction Map Definitions, 40](#)
- [Previewing PowerExchange Data in Designer, 44](#)
- [Group Source Processing in PowerExchange, 47](#)
- [Multiple-Record Writes to Nonrelational Targets, 50](#)
- [Source Qualifier Transformations, 52](#)
- [Lookup Transformations, 53](#)
- [Stored Procedure Transformations, 56](#)

Working with Mappings Overview

A mapping is a set of source and target definitions linked by transformation objects that define the rules for data transformation. Mappings represent the data flow between sources and targets.

Source and target definitions represent metadata for sources and targets. When you create a source definition, its structure differs depending on the type of source it represents. The source qualifier for a source definition also differs in structure depending on type of source definition.

After you create a source or target definition, you can include it in a mapping to extract data from the source or load data to the target. You can extract source data in batch, change, or real-time mode.

With CDC mappings, it is generally necessary to have multiple mappings: a batch mapping to materialize the target tables from the source tables in preparation for CDC and the CDC mapping itself which then uses extraction map sources for the source tables. To minimize the effort in creating these mappings, create any business rules applicable to both the batch and CDC sessions in mapplets.

Source and Target Definitions

To import relational and nonrelational sources from PowerExchange, in the Source Analyzer, click **Sources > Import from PowerExchange**.

To import relational and nonrelational targets from PowerExchange, in the Target Designer, click **Targets > Import from PowerExchange**.

In each case, the **Import from PowerExchange** dialog box appears. Certain fields in this dialog box appear for sources only, not for targets.

Additional input fields appear, based on the **Source Type** that you choose. After you create a source or target definition, you can edit it.

Working with Relational Source and Target Definitions

With PowerExchange Client for PowerCenter, you can create source definitions to read from, and target definitions to write to, the following relational databases:

- DB2 for z/OS
- DB2 for i5/OS
- DB2 for Linux, UNIX, and Windows
- Microsoft SQL Server
- Oracle

Working with DB2 Definitions

You can create a DB2 source definition in the following ways:

- Import the table definitions that PowerExchange retrieves from the DB2 catalog.
- Import DB2 or DB2 unload (DB2UNLD) data map definitions from PowerExchange.
- Import extraction map definitions for PowerExchange.
- Manually create a DB2 definition.

You can create a DB2 target definition in the following ways:

- Import the table definitions that PowerExchange retrieves from the DB2 catalog.
- Manually create a DB2 definition.
- Create a DB2 target definition from a DB2 source definition. In the Target Designer, drag a DB2 source definition to the workspace.

Note: If the source includes a LOB column, the source definition and source qualifier include a port for the ROWID column. The mapping does not use this port, and you can leave it unconnected to the target.

Tip: If your repository already contains DB2 definitions, you can use them to extract data from or load data to a DB2 table. However, the metadata definition must match the table structure of the DB2 table.

RELATED TOPICS:

- [“Importing Nonrelational Source Definitions” on page 33](#)
- [“Working with Extraction Map Definitions” on page 40](#)

Importing a DB2 Relational Source or Target Definition

You can connect to a DB2 database through a PowerExchange Listener or locally without a Listener to import DB2 metadata for creating PowerCenter source or target definitions. This procedure describes the import process for DB2 for z/OS, DB2 for i5/OS, or DB2 for Linux, UNIX, and Windows databases.

After you connect to the DB2 database, the Designer displays the available schemas and tables. You can filter this metadata by schema, table name, or both. Then select a table to create the source or target definition. When you import DB2 source or target definitions through a PowerExchange Listener or locally, the Designer imports the primary keys.

You can map DB2 tables in PowerExchange as either DB2 data maps or DB2 database unload (DB2UNLD) data maps for DB2 for z/OS sources. Import these types of data maps for DB2 sources in the same manner as nonrelational data map sources.

1. To import a DB2 source definition, click **Sources > Import from PowerExchange** and select a source type of DB2zOS, DB2i5OS, or DB2LUW.

To import a DB2 target definition, click **Targets > Import from PowerExchange** and select a source type of DB2zOS, DB2i5OS, or DB2LUW.

2. Enter connection information.

The following table describes the required and optional connection attributes:

Attribute	Required or Optional	Description
Location	Required	Name of the PowerExchange Listener (Node name from PowerExchange dbmover.cfg) on the system on which the database resides.
User Name	Required	A user name that has the authority to connect to the database. For connections to supported Linux, UNIX, or Windows platforms, if you have enabled PowerExchange LDAP user authentication and disabled relational pass-through authentication, the user name is the enterprise user name. For more information, see the <i>PowerExchange Reference Manual</i> .

Attribute	Required or Optional	Description
Password	Required	<p>A password for the specified user. Passwords that are less than nine characters in length are not validated.</p> <p>To avoid errors that might arise due to code page differences, Informatica recommends that the password includes only those characters that are allowed for passphrases.</p> <p>For a DB2 for i5/OS or DB2 for z/OS source or target, you can enter a valid PowerExchange passphrase instead of a password. 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. A passphrase can contain the following characters:</p> <ul style="list-style-type: none"> - Uppercase and lowercase letters - The numbers 0 to 9 - Spaces - The following special characters: ' - ; # \ , . / ! % & * () _ + { } : @ < > ? <p>Note: The first character is an apostrophe.</p> <p>Passphrases cannot include single quotation marks ('), double quotation marks ("), or currency symbols.</p> <p>On z/OS, the allowable characters in the IBM IRRPHREX exit do not affect the allowable characters in PowerExchange passphrases.</p> <p>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.</p> <p>To use passphrases, ensure that the PowerExchange Listener runs with a security setting of SECURITY=(1,N) or higher in the DBMOVER member. For more information, see "SECURITY Statement" in the <i>PowerExchange Reference Manual</i>.</p>
Multi-Record Datamaps	-	Select to list multi-record data maps. SEQ or VSAM sources only.
Source Type	Required	DB2zOS, DB2i5OS, or DB2LUW.
CDC Datamaps	-	Select to list CDC extraction maps - source only.
Network Timeout	Optional	Number of seconds that elapse during a network send or receive operation before PowerExchange ends the connection and issues a timeout error message.
Subsystem Id	Required	Name of the subsystem for connection. DB2zOS only.
Database Name	Required	Name of the database for connection. DB2i5OS and DB2LUW only.
Schema	Optional	Enter a schema name to filter the resulting data maps.
Map name	Optional	Enter a table name to filter the resulting data maps.
Respect Case	Optional	If selected, PowerExchange filters metadata by using the exact case of the Schema and Map name fields. Otherwise, PowerExchange ignores the case when searching for data maps.

- Optionally, define a filter criteria to narrow the list schemas and tables that are displayed.

To define a filter, enter a schema name, table name, or both. You can enter name patterns by using the following wildcard characters:

- * (asterisk). Represents one or more characters.
- ? (question mark). Represents one character.

For example:

- A* displays schemas and tables that begin with an A.
- *A displays schemas and tables that end with an A.
- *cust* displays schemas that contain "cust."

4. Click **Connect**.

The Designer displays the metadata to import in the **Selected Datamaps** box. If no tables are found, the Designer displays "No Data Found."

5. Select the table or tables that you want to import.

To select multiple tables, use the Shift key or Ctrl key or click **Select all**.

6. Click **OK**.

The source or target definition appears in the workspace.

RELATED TOPICS:

- ["Importing Nonrelational Source Definitions" on page 33](#)

Working with Microsoft SQL Server Definitions

You can create a Microsoft SQL Server source definition in the following ways:

- In PowerExchange, import table definitions from Microsoft SQL Server.
- Import extraction map definitions from PowerExchange.
- In the PowerCenter ODBC interface, import table definitions from Microsoft SQL Server.
- Manually create a Microsoft SQL definition.

You can create a Microsoft SQL Server target definition in the following ways:

- In PowerExchange, import table definitions from Microsoft SQL Server.
- In the PowerCenter ODBC interface, import table definitions from Microsoft SQL Server.
- Manually create a Microsoft SQL definition.

Tip: If your repository contains Microsoft SQL Server definitions, you can use them to extract data from or load data to a Microsoft SQL Server table. However, the metadata definition must match the table structure of the Microsoft SQL Server table.

RELATED TOPICS:

- ["Working with Extraction Map Definitions" on page 40](#)

Importing a Microsoft SQL Server Relational Source or Target Definition

You can connect to a Microsoft SQL Server database through a PowerExchange Listener, or locally without a Listener, to import SQL Server metadata. When you connect to the database, the Designer displays database schemas and tables. Before you connect to the database, you can filter the metadata the Designer displays by schema, table name, or by both values.

Select a table to create the source or target definition. When you import a Microsoft SQL Server definition through a PowerExchange Listener or locally, the Designer imports the primary keys.

To import a Microsoft SQL Server relational source or target definition:

1. To import a Microsoft SQL Server source definition, click **Sources > Import from PowerExchange** and select a source type of MSSQL.

To import a Microsoft SQL Server target definition, click **Targets > Import from PowerExchange** and select a source type of MSSQL.

2. Enter connection information.

The following table describes required and optional connection information:

Attribute	Required/ Optional	Description
Location	Required	Name of the PowerExchange Listener (Node name from PowerExchange dbmover.cfg) on the system on which the database resides.
User Name	Required	User ID for connection to the database.
Password	Required	Password associated with the User ID.
Multi-Record Datamaps	-	Select to list multi-record data maps. SEQ or VSAM sources only.
Source Type	Required	MSSQL.
CDC Datamaps	-	Select to list CDC extraction maps - source only
Network Timeout	Optional	Number of seconds that elapse during a network send or receive operation before PowerExchange ends the connection and issues a timeout error message.
Server Name	Required	SQL Server instance name.
Database Name	Required	Database name in the SQL instance specified.
Schema	Optional	Enter a schema name to filter the resulting data maps.
Map name	Optional	Enter a table name to filter the resulting data maps.
Respect Case	Optional	If selected, PowerExchange filters data maps by using the exact case of the Schema and Map name fields. Otherwise, PowerExchange ignores the case when searching for data maps.

3. Optionally, enter a filter to view particular schemas and tables from the database.

Enter a schema name, table name, or both values.

You can enter a filter condition to display schemas and tables that meet the filter condition. Use one of the following wildcard characters in the filter condition:

- * (asterisk). Represents one or more characters.
- ? (question mark). Represents one character.

Use the following syntax when entering a wildcard character:

- **Enter the filter condition as a prefix.** For example, enter A* to display schemas and tables that begin with an A.
- **Enter the filter condition as a suffix.** For example, enter *A to display schemas and tables that end with an A.
- **Enter the filter condition as a substring.** For example, enter *cust* to display schemas and tables that contain "cust."

4. Click **Connect**.

The Designer displays metadata to import in the **Selected Datamaps** box. If no tables are found, "No Data Found" is displayed.

5. Select the table you want to import.

- Hold down the Shift key to select blocks of tables.
- Hold down the Ctrl key to make non-contiguous selections within a schema.
- Use the Select all button to select all tables.
- Use the Select none button to clear all highlighted selections.

6. Click **OK**.

The source or target definition appears in the workspace.

Working with Oracle Definitions

You can create an Oracle source definition in the following ways:

- In PowerExchange, import table definitions from Oracle.
- Import extraction map definitions from PowerExchange.
- In the PowerCenter ODBC interface, import table definitions from Oracle.
- Manually create an Oracle source definition.

You can create an Oracle target definition in the following ways:

- In PowerExchange, import table definitions from Oracle.
- In the PowerCenter ODBC interface, import table definitions from Oracle.
- Manually create an Oracle source definition.

Tip: If your repository contains Oracle definitions, you can use them to extract data from or load data to an Oracle table. However, the metadata definition must match the table structure of the Oracle table.

RELATED TOPICS:

- ["Working with Extraction Map Definitions" on page 40](#)

Importing an Oracle Relational Source or Target Definition

You can connect to an Oracle database through a PowerExchange Listener, or locally without a Listener, to import Oracle metadata. When you connect to the database, the Designer displays database schemas and tables. Before you connect to the database, you can filter the metadata the Designer displays by schema, table name, or both values.

Select a table to create the source or target definition. When you import an Oracle source definition through a PowerExchange Listener, the Designer imports the primary keys.

To import an Oracle relational source or target definition:

1. To import an Oracle source definition, click **Sources > Import from PowerExchange** and select a source type of Oracle.

To import an Oracle target definition, click **Targets > Import from PowerExchange** and select a source type of Oracle.

2. Enter connection information.

The following table describes required and optional connection information:

Attribute	Required/ Optional	Description
Location	Required	Name of the PowerExchange Listener (Node name from PowerExchange dbmover.cfg) on the system on which the database resides.
User Name	Required	User name for connecting to the database. For sources or targets on supported Linux, UNIX, or Windows systems, if you have enabled PowerExchange LDAP user authentication and disabled relational pass-through authentication, the user name is the enterprise user name. For more information, see the <i>PowerExchange Reference Manual</i> .
Password	Required	Password associated with the user name.
Multi-Record Datamaps	-	Select to list multi-record data maps. SEQ or VSAM sources only.
Source Type	Required	ORACLE.
CDC Datamaps	-	Select to list CDC extraction maps - source only.
Network Timeout	Optional	Number of seconds that elapse during a network send or receive operation before PowerExchange ends the connection and issues a timeout error message.
TNS Name	Required	Oracle SID (instance name).
Schema	Optional	Enter a schema name to filter the resulting data maps.
Map name	Optional	Enter a table name to filter the resulting data maps.

3. Optionally, enter a filter to view particular schemas and tables from the database.

Enter a schema name, table name, or both values.

You can enter a filter condition to display schemas and tables that meet the filter condition. Use one of the following wildcard characters in the filter condition:

- * (asterisk). Represents one or more characters.
- ? (question mark). Represents one character.

Use the following syntax when entering a wildcard character:

- **Enter the filter condition as a prefix.** For example, enter A* to display schemas and tables that begin with an A.
- **Enter the filter condition as a suffix.** For example, enter *A to display schemas and tables that end with an A.

- **Enter the filter condition as a substring.** For example, enter *cust* to display schemas and tables that contain "cust."
4. Click **Connect**.
The Designer displays metadata to import in the **Selected Datamaps** box. If no tables are found, "No Data Found" is displayed.
 5. Select the table you want to import.
 - Hold down the Shift key to select blocks of tables.
 - Hold down the Ctrl key to make noncontiguous selections within a schema.
 - Use the Select all button to select all tables.
 - Use the Select none button to clear all highlighted selections.
 6. Click **OK**.
The source or target definition displays in the workspace.

Editing Relational Source and Target Definitions

You can edit relational source and target definition to make minor changes to the definition. For example, you can:

- Modify column names.
- Modify column datatypes.
- Add or delete columns.
- Modify column key relationships.
- Add a description of the definition.

Note: If you use the relational source in a CDC session, you do not need to add the DTL__CAPXACTION field nor do you need to include an Update Strategy transformation. PWXPC will automatically include the DTL__CAPXACTION column in its SELECT statement for CDC sources. It then uses the value of the DTL__CAPXACTION to construct the appropriate SQL statement (INSERT, UPDATE, or DELETE).

Working with Nonrelational Source and Target Definitions

With the PowerExchange Client for PowerCenter, you can create source definitions and target definitions for the following nonrelational source to target types:

- Adabas
- Datacom - source only
- DB2 data maps (DB2MAP) - source only
- DB2 for z/OS unload files (DB2UNLD) - source only
- IDMS - source only
- IMS
- SEQ
- VSAM

Importing Nonrelational Source Definitions

You can connect to PowerExchange locally or to a PowerExchange Listener to import a nonrelational data maps to create source definitions or a data mapped DB2 definition.

When you connect to PowerExchange, the Designer displays data map schemas and tables for the source type specified. Before you connect to PowerExchange, you can filter this information based on a schema name, data map name, or both. Then select a data map to create the source definition.

The following procedure imports single- and multi-record data maps to create source definitions. You can import multiple-record data maps for IMS unload data sets, VSAM data sets, and sequential files. For multiple-record data maps, select the **Multi-Record Datamaps** attribute.

1. Click **Sources > Import from PowerExchange**.
2. In the **Import from PowerExchange** dialog box, enter information for importing a nonrelational data map.

The following table describes the required and optional attributes:

Attribute	Required or Optional	Description
Location	Required	Name of the node where the PowerExchange Listener runs, as specified in a NODE statement in the PowerExchange DBMOVER configuration file that is on the system where the database resides.
User Name	Required	A user name that has the authority to connect to the database. For sources on supported Linux, UNIX, or Windows systems, if you have enabled PowerExchange LDAP user authentication, the user name is the enterprise user name. For more information, see the <i>PowerExchange Reference Manual</i> .

Attribute	Required or Optional	Description
Password	Required	<p>A password for the specified user. Passwords that are less than nine characters in length are not validated.</p> <p>To avoid errors that might arise due to code page differences, Informatica recommends that the password includes only those characters that are allowed for passphrases.</p> <p>For nonrelational sources on z/OS, and for sequential source files on i5/OS for bulk data movement, you can enter a valid PowerExchange passphrase instead of a password. 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. A passphrase can contain the following characters:</p> <ul style="list-style-type: none"> - Uppercase and lowercase letters - The numbers 0 to 9 - Spaces - The following special characters: ' - ; # \ , . / ! % & * () _ + { } : @ < > ? <p>Note: The first character is an apostrophe.</p> <p>Passphrases cannot include single quotation marks ('), double quotation marks ("), or currency symbols.</p> <p>On z/OS, the allowable characters in the IBM IRRPHREX exit do not affect the allowable characters in PowerExchange passphrases.</p> <p>Note: On z/OS, valid RACF passphrase can be up to 100 characters in length. PWXPC truncates passphrases longer than 100 characters when passing them to RACF for validation.</p> <p>To use passphrases, ensure that the PowerExchange Listener runs with a security setting of SECURITY=(1,N) or higher in the DBMOVER member. For more information, see "SECURITY Statement" in the <i>PowerExchange Reference Manual</i>.</p>
Multi-Record Datamaps	Optional	<p>Select this option to list multiple-record data maps, which are available for IMS, SEQ, and VSAM sources only. For more information, see "Group Source Processing in Bulk Data Movement Sessions with Nonrelational Data Sources" on page 47.</p> <p>Caution: When you try to import a very large multiple-record data map for a nonrelational data source into PowerCenter Designer, the import might fail. In this case, instead of creating a multiple-record data map, create a separate data map for each source record.</p>
Use Sequence Fields	Optional	<p>Select this option to generate sequence fields. IMS unload sources only. You can select this option only if you also select Multi-Record Data Maps. For more information, see "Multiple-Record Writes to Nonrelational Targets" on page 50.</p>
Source Type	Required	<p>Select one of the following source types: ADABAS, DATACOM, DB2MAP, DB2UNLD, IDMS, IMS, SEQ, or VSAM.</p>
CDC Datamaps	-	<p>Select this option to list CDC extraction maps - source only.</p>
Network Timeout	Optional	<p>The number of seconds that elapse during a network send or receive operation before PowerExchange ends the connection and issues a timeout error message.</p>
Schema	Optional	<p>A schema name to use for filtering the data maps.</p>

Attribute	Required or Optional	Description
Map name	Optional	A data map name to use for filtering the data maps.
Respect Case	Optional	Select this option to have PowerExchange use case-sensitive filtering of the data maps based on the values in the Schema and Map name fields. When this option is not selected, PowerExchange ignores case when searching for data maps.
Selected Datamaps	-	Lists the available data maps for the connection, source type, and filter details that you entered.

- Optionally, define a filter to narrow the list of schemas and tables.

To define a filter condition, enter a schema name, a table name, or both. You can enter name patterns by using the following wildcard characters:

- * (asterisk). Represents one or more characters.
- ? (question mark). Represents one character.

For example:

- A* displays schemas and tables that begin with an A.
- *A displays schemas and tables that end with an A.
- *cust* displays schemas that contain "cust."

- Click **Connect**.

The Designer displays the metadata for the data maps in the **Selected Datamaps** box. If no data maps are found, the Designer displays "No Data Found."

- Select the data map or data maps that you want to import.

To make multiple selections, use the Shift key or Ctrl key or click **Select all**.

- Click **OK**.

The source definitions are displayed. The Designer uses the data map names as the names of the source definition.

Caution: If you select more than one source object in the **Import from PowerExchange** dialog box and do not select **Multi-Record Datamaps**, the resulting target objects might have no columns.

To avoid this problem, select **Respect Case** in the **Import from PowerExchange** dialog box.

Importing Nonrelational Target Definitions

You can connect to PowerExchange locally or to a PowerExchange Listener to import a nonrelational target definition.

When you connect to PowerExchange, the Designer displays data map schemas and tables for the specified target type. Before you connect to PowerExchange, you can filter the metadata by schema, data map name, or both. Then select a data map to create the target definition.

- To import a nonrelational target definition, click **Targets > Import from PowerExchange** and select the target type.

The dialog box and parameters displayed are the same for each nonrelational target type.

- Enter connection information.

The following table describes required and optional connection attributes:

Attribute	Required or Optional	Description
Location	Required	Name of the PowerExchange Listener (Node name from PowerExchange dbmover.cfg) on the system on which the database resides.
User Name	Required	A user name that has the authority to connect to the database. For targets on supported Linux, UNIX, or Windows systems, if you have enabled PowerExchange LDAP user authentication, the user name is the enterprise user name. For more information, see the <i>PowerExchange Reference Manual</i> .
Password	Required	<p>A password for the specified user. Passwords that are less than nine characters in length are not validated.</p> <p>To avoid errors that might arise due to code page differences, Informatica recommends that the password includes only those characters that are allowed for passphrases.</p> <p>For nonrelational targets on z/OS, and for sequential target files on i5/OS for bulk data movement, you can enter a valid PowerExchange passphrase instead of a password. 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. A passphrase can contain the following characters:</p> <ul style="list-style-type: none"> - Uppercase and lowercase letters - The numbers 0 to 9 - Spaces - The following special characters: ' - ; # \ , . / ! % & * () _ + { } : @ < > ? <p>Note: The first character is an apostrophe.</p> <p>Passphrases cannot include single quotation marks ('), double quotation marks ("), or currency symbols.</p> <p>On z/OS, the allowable characters in the IBM IRRPHREX exit do not affect the allowable characters in PowerExchange passphrases.</p> <p>Note: On z/Os, valid RACF passphrase can be up to 100 characters in length. PWXPC truncates passphrases longer than 100 characters when passing them to RACF for validation.</p> <p>To use passphrases, ensure that the PowerExchange Listener runs with a security setting of SECURITY=(1,N) or higher in the DBMOVER member. For more information, see "SECURITY Statement" in the <i>PowerExchange Reference Manual</i>.</p>
Multi-Record Datamaps	Optional	Select to list multi-record data maps. IMS unload targets only. For more information, see "Multiple-Record Writes to Nonrelational Targets" on page 50 .
Use Sequence Fields	Optional	Select to generate sequence fields in the target. IMS unload targets only. You can select this option only if you also select Multi-Record Data Maps . For more information, see "Multiple-Record Writes to Nonrelational Targets" on page 50 .
Source Type	Required	Select one of the following target types: ADABAS, IMS, SEQ, or VSAM.
Network Timeout	Optional	Number of seconds that elapse during a network send or receive operation before PowerExchange ends the connection and issues a timeout error message.

Attribute	Required or Optional	Description
Schema	Optional	Enter a schema name to filter the resulting data maps.
Map name	Optional	Enter a data map name to filter the resulting data maps.
Respect Case	Optional	If selected, PowerExchange filters data maps by using the exact case of the Schema and Map name fields. Otherwise, PowerExchange ignores the case when searching for data maps.
Selected Datamaps	n/a	Lists the available data maps for the connection, database and filter details that you entered.

- Optionally, define a filter to narrow the list of schemas and tables.

To define a filter condition, enter a schema name, a table name, or both. You can enter name patterns by using the following wildcard characters:

- * (asterisk). Represents one or more characters.
- ? (question mark). Represents one character.

For example:

- A* displays schemas and tables that begin with an A.
- *A displays schemas and tables that end with an A.
- *cust* displays schemas that contain "cust."

- Click **Connect**.

The Designer displays the metadata for the data maps to import in the **Selected Datamaps** box. If no data maps are found, the Designer displays "No Data Found."

- Select the data map or data maps that you want to import.

To make multiple selections, use the Shift key or Ctrl key or click **Select all**.

- Click **OK**.

The target definitions are displayed. The Designer uses the data map names as the names of the target definition.

Caution: If you select more than one target object in the **Import from PowerExchange** dialog box and do not select **Multi-Record Datamaps**, the resulting target objects might have no columns.

To avoid this problem, select **Respect Case** in the **Import from PowerExchange** dialog box.

Viewing Nonrelational Source and Target Definition Details

Nonrelational source and target definitions contain information about the data map structure on the Attributes tab. Additional metadata information is included on the Metadata Extensions tab.

Viewing Data Map Information

The Attributes tab of a nonrelational source or target definition lists each field in the definition. It also lists the attributes of each field and their values from the data map used to create the definition. The attributes include information such as the name of the base record to which the field belongs.

The following table lists the attributes the Attributes tab displays for each field in a nonrelational definition:

Attribute Name	Description
column_name	Name of the field in the data map.
base_rec	Name of the record to which the field belongs. This corresponds to name of the group the field belongs to in the source definition.
base_fld	Name of the base record and table field name in the following format: <Base_Field_Name>:<Table_Field_Name>
base_fld_tpe	PowerExchange datatype of the field.
base_fld_offset	Offset value from which the field starts. For example, if the value is 5, the field starts at the fifth position. You determine the offset value of each field based on the order of fields in the data map.

Viewing Metadata Extensions

Nonrelational source and target definitions have metadata extensions that displays additional information from the data map used to create the definition. With VSAM and sequential definitions, the File Name extension field can be modified.

You can override the data map name and schema name in the session properties.

The following table describes the extensions on the Metadata Extensions tab for a nonrelational definition:

Extension Name	Description
Access Method	Method that you specified in the data map to access the source database: <ul style="list-style-type: none">- A = Adabas- D = IMS DL1- E = VSAM ESDS- I = IDMS- K = VSAM KSDS- O = IMS ODBA- N = VSAM RRDS- S = sequential (SEQ)- W = DB2 unload file (DB2UNLD)- X = Datacom- Z = DB2 data map
comments	Any comments.
File Name	Name of the data set (sequential or VSAM only).
Hierarchy Structure	For source and target definitions for which you selected Use Sequence Fields , the XML that describes the hierarchy structure of the data. Click the arrow in the Value column to display the XML. For more information, see "Hierarchy Structure" on page 51 .

Extension Name	Description
Map Name	Name of the data map.
Schema Name	Name of the schema on which you based the data map.
Use Sequence Fields	Whether to generate sequence fields for the source and target definitions. For more information, see “Multiple-Record Writes to Nonrelational Targets” on page 50 .

RELATED TOPICS:

- [“Editing Metadata Extensions ” on page 39](#)
- [“Configuring Batch Mode Sessions for Nonrelational Sources” on page 131](#)

Editing Nonrelational Source and Target Definitions

You can edit nonrelational source and target definitions to make minor changes to the definition. For example, you can:

- Modify a column datatype
- Modify the owner name
- Modify column key relationships
- Add or delete columns
- Add a description of the definition
- Create metadata extensions

Note: If you use the nonrelational source in a CDC session you do not need to add the DTL__CAPXACTION field nor do you need to include an Update Strategy transformation. PWXPC will automatically include the DTL__CAPXACTION column in its SELECT statement for CDC sources. It then uses the value of the DTL__CAPXACTION to construct the appropriate SQL statement (INSERT, UPDATE, or DELETE).

Editing Metadata Extensions

With VSAM and sequential definitions, the File Name extension field can be modified. All other field populated by PWXPC when a data map is imported cannot be changed.

By default, the File Name field contains the value from the PowerExchange data map File Name field which contains the VSAM data set name or the sequential file name.

Note: Any VSAM and sequential definitions created prior to PowerCenter 8.5 will have a blank File Name. If desired, this field can be populated manually or by re-importing the data map.

The following procedure explains how to manually update the File Name field to add or change the file name in a VSAM or sequential definition.

To edit metadata extensions:

1. Double-click the source or target definition in the workspace.
2. Select the Metadata Extensions tab.
PWXPC populates the **File Name** field with the File Name for the PowerExchange data map.
3. Click in the **Value** field and select the down arrow.
The Edit Metadata Extension Value panel appears, allowing you to enter or change the file name.

4. Click **OK**.
5. Click **OK** to end Edit Tables.

You can also re-import the PowerExchange data map to automatically populate the **File Name** field with the information contained in the data map.

Working with Extraction Map Definitions

When creating mappings for change data capture (CDC), import the extraction map rather than the source table metadata.

Extraction maps provide the following benefits:

- Source extraction maps can contain a subset of the source columns to reduce capture processing.
- Extraction maps contain the PowerExchange-generated columns, such as DTL__CAPXTIMESTAMP and DTL__CAPXACTION, and DTL__BI, which provided information needed for CDC.

To import an extraction map, connect to source location where the extraction maps reside, either locally or through a PowerExchange Listener.

After you connect to PowerExchange, the Designer displays the schemas and tables in the extraction map. You can filter the extraction maps by schema name, map name, or by both. Then select an extraction map to create the source definition.

1. To import an extraction map for a CDC source definition, click **Sources > Import from PowerExchange** and select **CDC Datamaps**. Then select the source type.

The dialog box and parameters displayed are the same for each nonrelational source type.

2. Enter connection information.

The following table describes required and optional connection attributes:

Attribute	Required or Optional	Description
Location	Required	Node name of the PowerExchange Listener, as specified in a NODE statement in the PowerExchange DBMOVER configuration file on the system where the database resides.
User Name	Required	A user name that has the authority to connect to the database. For sources on supported Linux, UNIX, or Windows systems, if you have enabled PowerExchange LDAP user authentication, the user name is the enterprise user name. For more information, see the <i>PowerExchange Reference Manual</i> .

Attribute	Required or Optional	Description
Password	Required	<p>A password for the specified user. Passwords that are less than nine characters in length are not validated.</p> <p>To avoid errors that might arise due to code page differences, Informatica recommends that the password includes only those characters that are allowed for passphrases.</p> <p>For access to i5/OS or z/OS, you can enter a valid PowerExchange passphrase instead of a password. 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 when using PWXPC. A passphrase can contain the following characters:</p> <ul style="list-style-type: none"> - Uppercase and lowercase letters - The numbers 0 to 9 - Spaces - The following special characters: ' - ; # \ , . / ! % & * () _ + { } : @ < > ? <p>Note: The first character is an apostrophe.</p> <p>Passphrases cannot include single quotation marks ('), double quotation marks ("), or currency symbols.</p> <p>On z/OS, the allowable characters in the IBM IRRPHREX exit do not affect the allowable characters in PowerExchange passphrases.</p> <p>Note: On z/OS, a valid RACF passphrase can be up to 100 characters in length. PWXPC truncates passphrases longer than 100 characters when passing them to RACF for validation.</p> <p>To use passphrases, ensure that the PowerExchange Listener runs with a security setting of SECURITY=(1,N) or higher in the DBMOVER member. For more information, see "SECURITY Statement" in the <i>PowerExchange Reference Manual</i>.</p>
Multi-Record Datamaps	-	Select this option to list multi-record data maps. SEQ or VSAM sources only.
Source Type	Required	<p>One of the following source types:</p> <ul style="list-style-type: none"> - ADABAS - DATACOM - DB2zOS - DB2i5OS - DB2LUW - IDMS - IMS - MSSQL - ORACLE - VSAM
CDC Datamaps	Required	Select this option to list CDC extraction maps.
Network Timeout	Optional	The number of seconds that must elapse during a network send or receive operation before PowerExchange ends the connection and issues a timeout error message.
Schema	Optional	A schema name that is used to filter the resulting data maps.

Attribute	Required or Optional	Description
Map name	Optional	Enter a data map name to filter the data maps.
Respect Case	Optional	Select this option to have PowerExchange filter data maps by using the exact case of the values in the Schema and Map name fields. When this option is not selected, PowerExchange ignores the case when searching for data maps.

- Optionally, define a filter criteria to narrow the list schemas and tables that are displayed.
To define a filter condition, enter a schema name, a table name, or both. You can enter name patterns by using the following wildcard characters:
 - * (asterisk). Represents one or more characters.
 - ? (question mark). Represents one character.
 For example:
 - A* displays schemas and tables that begin with an A.
 - *A displays schemas and tables that end with an A.
 - *cust* displays schemas that contain "cust."
- Click **Connect**.
The Designer displays metadata to import in the **Selected Datamaps** box. If no tables are found, "No Data Found" displays. The extraction maps shown will be filtered based on the source type specified.
- Select the extraction map or maps that you want to import.
To select multiple extraction maps, use the Shift key or Ctrl key or click **Select all**.
- Click **OK**.
The source definition appears in the workspace. The Designer uses the extraction map name as the name of the source definition.

Note: In PowerCenter Designer, if you use the **Import from PowerExchange** option to import a DB2 for Linux, UNIX, and Windows extraction map for a source that has long table and column names, some column names might not be displayed or long column names might be truncated to 74 characters without any warning or error message. Alternatively, if you use the **Import from PowerExchange** option to import table definitions from the DB2 catalog for a source or target table that has table or column names longer than 80 characters, Designer displays a message that indicates the table name or column name is too long and the import fails.

Viewing Extraction Map Definition Details

Extraction map definitions contain information about the data map structure on the Attributes tab. They also contain metadata extensions on the Metadata Extensions tab.

Viewing Extraction Map Information

The Attributes tab of an extraction map definition lists each field in the source definition. It also lists the attributes of each field and their values based on the extraction map from which you imported the source definition. The attributes include information such as the name of the base record to which the field belongs.

The following table describes the attributes that the Attributes tab displays for each field in a source definition:

Attribute Name	Description
column_name	Name of the field in the extraction map.
base_rec	blank.
base_fld	blank
base_fld_tpe	blank.
base_fld_offset	blank

Viewing Metadata Extensions in Extraction Maps

Extraction map definitions have metadata extensions that display information about the data map on which you based the source definition. You cannot modify these metadata extensions. However, you can override the data map name and schema name in the session properties.

The following table describes the extensions on the **Metadata Extensions** tab for an extraction map:

Extension Name	Description
Access Method	Method you specified in the data map to access the source database: <ul style="list-style-type: none"> - A = Adabas - B = DB2 for z/OS and DB2 for i5/OS - D = IMS - E = VSAM ESDS - I = IDMS - K = VSAM KSDS - L = MSSQL - N = VSAM RRDS - P = Oracle - V = DB2 for Linux, UNIX, and Windows - X = Datacom
comments	Any comments.
Map Name	Name of the extraction map.
Original Name	Original table name in relational database or PowerExchange data map.
Original Schema	Original schema or owner name in relational database or PowerExchange data map.
Schema Name	Name of the schema for the extraction map.

RELATED TOPICS:

- [“Configuring Batch Mode Sessions for Nonrelational Sources” on page 131](#)

Editing an Extraction Map Definition

You can edit extraction maps to make minor changes to them, for example:

- Modify a column datatype.
- Modify the owner name.
- Add or delete columns.
- Add a description of the extraction map.
- Create metadata extensions.

When you use extraction maps, you do not need to add the DTL__CAPXACTION field nor do you need to include an Update Strategy transformation. PWXPC will automatically include the DTL__CAPXACTION column in its SELECT statement for CDC sources. It then uses the value of the DTL__CAPXACTION to construct the appropriate SQL statement (INSERT, UPDATE, or DELETE).

Warning: Changing column information in the extraction map could result in failures in the session or workflow during the extraction process.

Previewing PowerExchange Data in Designer

You can preview data in the Designer for any valid relational and nonrelational source or target definitions. You can also preview change data for extraction map source definitions. You cannot preview data for multi-record data map source definitions.

Previewing data allows you to see the source or target data. The data that is shown depends upon the database type of the source or target definition. There are three database types for PowerExchange source and target definitions:

- Relational, such as DB2 for DB2 metadata.
- Nonrelational, such as PWX_VSAM_NRDB2 for VSAM data maps.
- Extraction Maps, such as PWX_DB2390_CDC for DB2 CDC data maps.

You can preview source or target data in the following Designer tools:

- **Source Analyzer.** Preview source data in the Source Analyzer after you import the source.
- **Target Designer.** Preview target data in the Target Designer after you import a target.
- **Mapplet Designer.** Preview source data in the Mapplet Designer while creating a mapplet.
- **Mapping Designer.** Preview source and target data in the Mapping Designer while creating a mapping.

Previewing PowerExchange Relational Source or Target Data

Preview data for relational source or target tables based on the source or target relational definitions.

Note: You must be able to connect to PowerExchange on the source or target system. To preview relational data, you must use an ODBC data source to connect to PowerExchange.

1. Select a relational source or target definition in the workspace.

2. Right-click the source or target definition in the workspace and click **Preview Data**.
The **Preview Data** dialog box appears.
3. Select an ODBC data source name.
You can add an ODBC data source by clicking the **Browse** button.
4. If necessary, enter **Username** and **Password** values.
For PowerExchange sources and targets on i5/OS or z/OS, a user name and password are required only if connecting to a PowerExchange Listener that is configured with a PowerExchange security setting of SECURITY=(1,x) or (2,x) in the DBMOVER configuration file.
To connect to sources or targets on i5/OS or z/OS, you can enter a valid PowerExchange passphrase instead of a password. An i5/OS passphrase can be from 9 to 31 characters in length. A z/OS passphrase can be from 9 to 79 characters in length when using ODBC. A passphrase can contain uppercase and lowercase letters, numbers, spaces, and the following special characters:
 ' - ; # \ , . / ! % & * () _ + { } : @ | < > ?
Note: The first character is an apostrophe.
5. Enter a table owner name.
6. Enter the number of rows that you want to preview.
Default is 100. The **Preview Data** dialog box can display up to 500 rows and up to 65,000 columns.
7. Click **Connect**.
The table contents appears in the Preview Data dialog box.
8. To change the number of rows that are displayed, enter another number of rows and click **Refresh**.
9. When you are done previewing data, click **Close**.

Previewing PowerExchange Nonrelational Source or Target Data

Preview data from nonrelational sources or targets based on the source or target definitions. For DB2, the preview process displays data from the source DB2 table.

To preview data, you must be able to connect to PowerExchange on the source or target system. PowerCenter uses PWXPC to connect to PowerExchange for nonrelational sources or targets.

1. Select a nonrelational source or target definition in the workspace.
2. Right-click the source or target definition in the workspace and click **Preview Data**.
The **Preview Data** dialog box appears.
3. Select a **Location** name.
The location names are retrieved from the NODE statements in the dbmover.cfg file on the Designer system. To add locations, add NODE statements to the dbmover.cfg file.
4. If necessary, enter **Username** and **Password** values.
For PowerExchange sources and targets on i5/OS and z/OS, a user name and password is required only if connecting to a PowerExchange Listener that is configured with a PowerExchange security setting of SECURITY=(1,x) or (2,x) in the DBMOVER configuration file.
To connect to sources or targets on z/OS, or to connect to sequential source or target files on i5/OS for bulk data movement, you can enter a valid PowerExchange passphrase instead of a password. 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 when using a PWXPC connection. A passphrase can contain uppercase and lowercase letters, numbers, spaces, and the following special characters:

' - ; # \ , . / ! % & * () _ + { } : @ | < > ?

The first character is an apostrophe.

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.

5. In the **Schema** field, change the schema name from the data map, if necessary.
6. In the **Network Timeout** field, enter the number of seconds that must elapse during a network send or receive operation before PowerExchange ends the connection and issues a timeout error message. Default is 180.
7. Enter the number of rows that you want to preview.
Default is 10. The **Preview Data** dialog box can display up to 500 rows and up to 65,000 columns.
8. Click **Connect**.
The source or target contents appears in the **Preview Data** dialog box.
9. To view more data, click **More**.
10. To change the number of rows that are displayed, enter another number of rows and click **Refresh**.
11. When you are done previewing data, click **Close**.

Previewing PowerExchange Change Data

Preview data based on extraction map definitions (CDC Datamaps) to display change data from the change stream.

You must be able to connect to PowerExchange on the source platform to preview data. PowerCenter uses PWXPC to connect to PowerExchange for previewing CDC data based on extraction maps.

1. Select an extraction map source definition in the workspace.
2. Right-click the source definition in the workspace and click **Preview Data**.

The **Preview Data** dialog box appears.

3. Select a **Location** name.

The location names are retrieved from the NODE statements in the dbmover.cfg configuration file on the Designer platform. To add additional locations, add NODE statements to the dbmover.cfg file.

4. If necessary, enter **Username** and **Password** values.

For PowerExchange sources and targets on z/OS, a user name and password are required only if you are connecting to a PowerExchange Listener that is configured with a PowerExchange security setting of SECURITY=(1,x) or (2,x) in the DBMOVER configuration file.

To connect to sources on i5/OS or z/OS, you can enter a valid PowerExchange passphrase instead of a password. 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 when using a PWXPC connection. A passphrase can contain uppercase and lowercase letters, numbers, spaces, and the following special characters:

' - ; # \ , . / ! % & * () _ + { } : @ | < > ?

The first character is an apostrophe.

Note: On z/OS, a valid RACF passphrase can be up to 100 characters in length. PowerExchange truncates z/OS passphrases longer than 100 characters when passing them to RACF for validation.

5. In the **Schema** field, change the schema name from the extraction map, if necessary.

6. In the **Network Timeout** field, enter the number of seconds that must elapse during a network send or receive operation before PowerExchange ends the connection and issues a timeout error message. Default is 180.
7. Select **Real Time** or **Change**.
The **Real Time** option extracts data in real time from the change stream by using the earliest starting restart point. The **Change** option extracts data from condense files by using the earliest restart point.
8. Enter the number of rows that you want to preview.
Default is 10. The **Preview Data** dialog box can display up to 500 rows and up to 65,000 columns.
9. Click **Connect**.
The source contents appears in the **Preview Data** dialog box.
10. To return more data, click **More**.
When the end of log is reached, the **More** button becomes unavailable.
11. To change the number of rows to preview, enter another number of rows and click **Refresh**.
12. When you done previewing data, click **Close**.

RELATED TOPICS:

- [“Default Restart Points” on page 165](#)

Group Source Processing in PowerExchange

With group source processing, PowerExchange reads data from the same physical source in a single pass. This processing enhances throughput and reduces resource consumption by eliminating multiple reads of the source data.

PowerExchange and PWXPC use group source processing when possible in CDC sessions for all data sources and in bulk data movement sessions for nonrelational data sources.

For CDC data, PowerExchange uses group source processing to read changes for all tables defined in a mapping from the change stream in a single pass. For nonrelational bulk data, PowerExchange uses group source processing to read all records in a single multi-group source qualifier in a mapping.

Group Source Processing in Bulk Data Movement Sessions with Nonrelational Data Sources

If you use PWXPC connections for bulk data movement operations, PowerExchange uses group source processing for the following nonrelational data sources:

- IMS unload data sets
- Sequential data sets and flat files
- VSAM data sets

Source Definitions for Sources with Multiple Record Types

You create a nonrelational source definition by importing a data map from a PowerExchange Listener. For data sources with multiple record types, the PowerExchange data map can define a record and a table for each unique record type. The table represents a relational view of the related record.

Nonrelational source definitions present the data map metadata in groups. Each group represents a table in the data map and contains metadata for the fields in the table.

For IMS, VSAM, and sequential or flat file data sources, use Designer to import data maps that define multiple record types to create one or more PowerCenter source definitions. To import a multiple-record data map, select **Multi-Record Datamaps** in the **Import from PowerExchange** dialog box.

To create a source definition for a single record, import a single table from the data map. The source definition then contains only a single group. To create a source definition that includes all record types, import the entire multiple-record data map. The source definition then contains a group for each table in the data map. Each group contains metadata for the fields in a table.

The following figure shows a nonrelational source definition for a VSAM data map that contains multiple tables representing multiple records in the VSAM file:

Name	Datatype	Length
V07A_RECORD_LAYOUT		
V07A_DATAGEN_PFX	CHAR	8
V07A_RECORD_KEY	UZONED	10
V07A_RECORD_TYPE	CHAR	2
V07A_CHAR_1	CHAR	10
V07A_CHAR_2	CHAR	10
V07A_CHAR_3	CHAR	10
V07B_RECORD_LAYOUT		
V07B_DATAGEN_PFX	CHAR	8
V07B_RECORD_KEY	UZONED	10
V07B_RECORD_TYPE	CHAR	2
V07B_PACK_1	PACKED	8
V07B_PACK_2	PACKED	8
V07B_PACK_3	PACKED	8
V07B_ZONE_1	ZONED	5
V07B_ZONE_2	ZONED	5
V07B_ZONE_3	ZONED	5
V07C_RECORD_LAYOUT		
V07C_DATAGEN_PFX	CHAR	8
V07C_RECORD_KEY	UZONED	10
V07C_RECORD_TYPE	CHAR	2
V07C_CDY4MMDD	UZONED	8
V07C_CDYYMMDD	UZONED	6
V07C_CDMMDYY	UZONED	6
V07C_SDMMDY4	UZONED	10
V07D_RECORD_LAYOUT		
V07D_DATAGEN_PFX	CHAR	8
V07D_RECORD_KEY	UZONED	10
V07D_RECORD_TYPE	CHAR	2
V07D_ZONE_DEC_1	ZONED	10
V07D_ZONE_DEC_2	ZONED	10
V07D_PACKED_1	PACKED	8

1 2

1. A section represents a table in the PowerExchange data map.
2. Group name.

In this example, the source definition contains four groups: V07A_RECORD_LAYOUT, V07B_RECORD_LAYOUT, V07C_RECORD_LAYOUT and V07D_RECORD_LAYOUT.

Group Source Processing and Multiple-Record Data Maps

You can use group source processing with a PowerExchange multiple-record data map for a nonrelational source with multiple record types.

When you import the data map into PowerCenter, select the **Multi-Record Datamaps** option in the **Import from PowerExchange** dialog box. PowerCenter creates a source definition that includes all of the record types that are defined in the data map. When a bulk data movement session with a mapping that contains the source definition runs, PowerExchange uses group source processing to read all of the records in the source data set or file in a single pass.

If you do not select the **Multi-Record Datamaps** option in the **Import from PowerExchange** dialog box, PowerCenter creates a separate source definition for each table, or record, in the multiple-record data map.

When a session with a mapping that contains these single-record source definitions runs, PowerExchange reads the source data set of file once for each source definition and does not use group source processing.

Note: PWXPC establishes a connection to the source for each source definition to read source data.

If you import IMS data maps as multiple-record data maps, you can use the source definitions only to process IMS unload data sets. You cannot use multiple-record IMS source definitions to read all segments from an IMS database in a single pass. For bulk data movement sessions with IMS source databases, create mappings that contain a source definition for each segment in the IMS database.

Group Source Processing in CDC Sessions

When you use PWX CDC application connections to extract change data, PowerExchange uses group source processing and reads the change stream in a single pass for all source definitions in the mapping.

When you run a CDC session, PWXPC passes a source interest list that contains all of the sources. PowerExchange uses the source interest list to determine the sources for which to read data from the change stream. When PowerExchange encounters changes for a source in the interest list, it passes the change data to PWXPC. PWXPC then provides the change data to the appropriate source in the mapping.

To create source definitions in Designer that can be used to extract change data, import source metadata by using one of the following methods:

- Import a PowerExchange extraction map by using the Import from **PowerExchange** dialog box.
- Import the table definitions from relational databases, by using either the **Import from PowerExchange** dialog box or the **Import from Database** dialog box.

Restriction: To read change data for nonrelational sources, you must import extraction maps from PowerExchange.

Informatica recommends that you use extraction maps to create source definitions for all CDC sources. When you create source definitions from extraction maps, the mapping and session creation process is simpler for the following reasons:

- The source definition contains the extraction map name, which eliminates the need to provide it when you configure the session.
- The source definition contains the PowerExchange-defined CDC columns, which eliminates the need to add these columns to the source definition. The PowerExchange-defined columns include the change indicator and before image columns as well as the DTL__CAPX columns.

When you extract change data, PowerExchange uses group source processing for all source definitions in the mapping. All source definitions must be for the same data source type, such as DB2, IMS, VSAM, or Oracle. Do not include multiple data source types in the mapping. Otherwise, the session fails with message PWXPC_10080.

For example, you cannot run a CDC session that contains a mapping with both VSAM and IMS source definitions, even though the change stream is the same. To extract change data for both IMS and VSAM data sources, create a mapping and session for the VSAM sources and a separate mapping and session for the IMS sources. PowerExchange reads the change stream twice, once for the session with VSAM sources and once for the session with IMS sources.

If you create a workflow that contains multiple CDC sessions, PowerExchange uses a connection for each session, even if the sessions extract change data from the same change stream, such as the PowerExchange Logger for z/OS.

Multiple-Record Writes to Nonrelational Targets

During bulk data movement sessions, PowerExchange can use a multiple-record data map to read data from a nonrelational data source and write the multiple record types in a single pass to a nonrelational target. This process is called a *multiple-record write*.

When PowerExchange performs a multiple-record write, it preserves source sequencing information. To enable multiple-record writes with sequencing, select the **Multi-Record Datamaps** and **Use Sequence Fields** options in the **Import from PowerExchange** dialog box for both the source and target definition. When you select these options, PowerExchange generates primary key and foreign key fields for each group in the source and target definitions and also stores metadata about these record relationships in XML format.

You can perform multiple-record writes with the following types of sources and targets:

- Source and target IMS unload data sets on z/OS
- Any combination of the following types of nonrelational sources and targets:
 - Sequential or VSAM ESDS, KSDS, or RRDS data sets on z/OS
 - Files on Linux, UNIX, or Windows
 - Files on i5/OS

RELATED TOPICS:

- [“Considerations for Sessions that Perform Multiple-Record Writes” on page 149](#)
- [“Connection Attributes for Multiple-Record Writes” on page 126](#)

Group Processing with Sequencing

To perform multiple-record writes with sequencing, PowerExchange uses group source processing to read source data in a single pass and uses group target processing to write data to the target in a single pass.

PowerExchange generates sequence fields to pass metadata about the source record relationships to the target. After you enable multiple-record writes with sequencing for a PowerCenter workflow, the workflow can read the multiple-record source, use the generated sequence fields to preserve the sequencing information, and write data to the target in the same record sequence as in the source.

To determine the relationships among IMS record types, PowerExchange uses the segment information that you imported from an IMS DBD into the multiple-record data map. To determine the relationships between record types for sequential or VSAM sources, PowerExchange uses the record properties that you define in the data map for the source or target. These record properties include the parent record name, the name of the record that the current record redefines, and whether the record is a header or trailer. PowerExchange uses the segment information or record properties to generate primary and foreign keys and sequencing metadata in the source and target definitions.

When you run a PowerCenter session, PowerExchange uses the generated key values to reconstruct and write the data to the target in the correct sequence. PowerExchange maintains the data in a sequencing and queuing cache on the Integration Service node. When PowerExchange writes data to the target, it deletes the generated key fields and sends the data across the network in the correct sequence to the target file.

For group source processing with sequencing, PowerExchange performs the following functions when it reads the source data:

- Reads the data in a single pass
- Drops any complex tables in the data map from the source definition
- Adds primary-key and foreign-key columns to the source definition

- Generates sequencing metadata to represent the source hierarchy

For group target processing with sequencing, PowerExchange performs the following functions when it writes the target data:

- Resequences the records in the cache.
- Discards sequence data fields after sequencing the records and before writing them to the target.
- Writes the data to the target in a single pass.

Sequencing Support in Source and Target Definitions

If you enable multiple-record write operations with sequencing when you import a multiple-record data map into PowerCenter, the resulting source or target definition includes special features to support record sequencing.

The following PowerCenter features support sequencing:

- The generated primary key and foreign key columns in the table definitions. The keys contain sequence numbers that uniquely identify each row and its parent.
- The following metadata extensions on the **Metadata Extensions** tab of the **Edit Tables** dialog box:
 - Use Sequence Fields. This metadata extension indicates that you selected the **Use Sequence Fields** option in the **Import from PowerExchange** dialog box.
 - Hierarchy Structure. This metadata extension is a segment of XML code that defines the hierarchy structure of the tables in a source or target definition.

Note: The **Database type** field on the **Table** tab of the **Edit Tables** dialog box contains different values for sequential source and target definitions. The database type for sequential source definitions is `PWX_VSAM`. The database type for sequential target definitions is `PWX_SEQ`.

Generated Primary Key and Foreign Key Columns

When you enable sequencing in a source definition, the source definition includes generated primary key and foreign key columns. These key columns are used to capture sequencing information from the source file and reconstruct the data in the same sequence in the target.

The source definition includes the following generated columns:

- `DTL_PK_tablename`. Contains a sequence number that is generated for each source table. This sequence number serves as the primary key.
- `DTL_FK_tablename`. Contains a sequence number that is generated for each table that corresponds to a non-root record. This sequence number serves as a foreign key.

The mapping must pass the generated sequence numbers, unaltered, to the corresponding generated columns on the target. The names and types of the generated columns must also remain unaltered. The names of the sequence fields are contained in the XML code that is defined in the Hierarchy Structure metadata extension.

Hierarchy Structure

The Hierarchy Structure metadata extension is a segment of XML code that defines the hierarchical structure of the tables in a source or target definition for which you enable sequencing. Developers can use this

information to write mapping logic that controls the relationships among tables in PowerCenter transformations.

The Hierarchy Structure includes the following information:

- Database type, data map, and schema
- Node definition for each record in the data map

Each node defines the following information:

- Node name
- Name of parent node (null string for root nodes)
- Horizontal and vertical position
- Record name
- Table name
- Column name of the generated primary key
- For non-root nodes, the column and name of the generated foreign key

The following lines of XML are an example of a Hierarchy Structure:

```
?<?xml version="1.0" encoding="UTF-16" standalone="no" ?>
<HIERARCHY DBTYPE="IMS" MAP="imsstudkey" SCHEMA="mrecims">

  <NODE NAME="STUDENT" PARENTNODE="" POSHOR="1" POSVERT="1">
    <RECORD NAME="STUDENT">
      <TABLE NAME="STUDENT">
        <COL_PK CANDIDATE="1" NAME="DTL_PK_STUDENT"/>
      </TABLE>
    </RECORD>
  </NODE>

  <NODE NAME="COURSE" PARENTNODE="STUDENT" POSHOR="1" POSVERT="2">
    <RECORD NAME="COURSE">
      <TABLE NAME="COURSE">
        <COL_PK CANDIDATE="1" NAME="DTL_PK_COURSE"/>
        <COL_FK CANDIDATE="1" NAME="DTL_FK_COURSE"/>
      </TABLE>
    </RECORD>
  </NODE>

</HIERARCHY>
```

Source Qualifier Transformations

Source Qualifier transformations determine the rows or records that the PowerCenter Integration Service reads from the source when you run a session.

When you add a relational or a flat file source definition to a mapping, you need to connect it to a Source Qualifier transformation. The type of source qualifier transformation depends on the source definition type:

- For relational source definitions, use a Source Qualifier transformation.
- For nonrelational source definitions, use an Application Multi-Group Source Qualifier transformation.

Source Qualifier Transformation Datatypes

The datatypes in Source Qualifier and Application Multi-Group Source Qualifier transformations are internal datatypes based on ANSI SQL-92 generic datatypes, which PowerCenter uses to move data across platforms.

When the PowerCenter Integration Service reads data from a source, it converts the data from the PowerExchange datatype to the transformation datatype. When you run a session, the PowerCenter Integration Service performs transformations based on the transformation datatypes. When writing data to a target, the PowerCenter Integration Service converts the data based on the datatypes in the target definition.

The transformation datatype for all ports in the Application Multi-Group Source Qualifier transformation are predefined. You cannot change the datatype for any of the fields in the Application Multi-Group Source Qualifier transformations.

Creating Source Qualifiers

By default, the PowerCenter Designer creates an Application Multi-Group Source Qualifier transformation when you add a nonrelational source definition to a mapping. Similarly, it creates a Source Qualifier transformation when you add a relational source definition to a mapping.

If you configure the PowerCenter Designer to manually create a source qualifier when you add a source definition to a mapping, you must manually connect the source qualifier to the source definition.

Editing Source Qualifier Transformations

After you create a Source Qualifier transformation, you can edit the transformation. You cannot edit an Application Multi-Group Source Qualifier transformation.

When you use a Source Qualifier transformation with the **Resume from last checkpoint** recovery strategy, you must select the **Output is Deterministic** option for the transformation. If you do not select this option, you will not be able to validate the CDC session in Workflow Manager.

Lookup Transformations

This section discusses considerations for lookup transformations with PWXPC for PowerExchange sources and targets.

Use a Lookup transformation in a mapping to look up data, through PowerExchange, in a nonrelational file or a relational table, view, or synonym. You can import a lookup definition for any flat file, nonrelational file, or relational table to which both the PowerCenter Client and PowerCenter Integration Service can connect.

Usage Considerations for Lookup Transformations

Review the following usage considerations if you use Lookup transformations with PowerExchange sources or targets:

- The Lookup transformation import process uses ODBC for nonrelational files and relational tables.
To use PWXPC to import definitions for nonrelational files or relational tables, first import the definitions in the **Import from PowerExchange** dialog box in either the Source Analyzer or Target Designer before configuring the lookup in the mapping.
- You can use PWXPC connections for lookups. The type of connection depends on the following criteria:
 - For lookups that involve relational tables, use the appropriate relational connection for the database type, such as PWX DB2zOS, PWX DB2i5OS, PWX DB2LUW, PWX Microsoft SQL Server, or PWX Oracle.
 - For lookups that involve nonrelational files, select the PWX NRDB Lookup connection.

- For NRDB lookups, the Lookup transformation property **Lookup Policy on Multiple Match** has no effect. The Lookup transformation always returns the first row that matches the lookup condition.
- For nonactive DB2 lookups, the Lookup transformation appends FETCH FIRST 2 ROWS ONLY to the SQL that it sends to PWXPC. In this way, DB2 optimizes its processing and returns a result set with a maximum of two rows. Nonactive DB2 lookups are lookups with a **Lookup Policy on Multiple Match** value of **Report Error**, **Use First Value**, **Use Last Value**, or **Use Any Value**.
- To use Lookup transformations with the **Resume from last checkpoint** recovery strategy, you must select either the **Lookup Source is Static** or **Lookup Cache Persistent** option for the transformation. If you do not select one of these options, you will not be able to validate the CDC session in Workflow Manager.
- To use Lookup transformations with IMS databases, carefully consider the fields that are used to perform the search of the IMS database. Concatenated keys (CCK) fields achieve the best performance with the least impact on the IMS database.
- To use Lookup transformations for targets in a CDC mapping, use special custom properties to ensure the change data stream is accessible across pipelines.
- When a PowerCenter workflow runs an IDMS netport job in Central Version (CV) mode, such as when performing a lookup on an IDMS data source, and delays occur in processing the source data or performing other lookups, the netport job might return IDMS error code 0069. This error occurs when no activity has occurred in the netport job for the length of the CV internal timeout and the bind run unit has expired.

To disable IDMS timeouts, specify the following IDMS System Generation parameters:

- CHKUSER TASK IS *number_of_tasks*
- EXTERNAL WAIT IS FOREVER

RELATED TOPICS:

- [“NRDB Lookup Relational Connections” on page 88](#)
- [“Configuring Lookups for IMS” on page 55](#)
- [“Configuring Lookups for CDC Data” on page 55](#)
- [“Configuring PowerExchange Connections” on page 61](#)

Limitations on Nonrelational Uncached Lookups

Nonrelational data sources that use the following PowerExchange access methods support cached lookups but not uncached lookups:

- DB2UNLD
- DB2390IMG
- ESDS
- IMSUNLD
- RRDS
- TAPE
- USER

Nonrelational data sources that use the following PowerExchange access methods support both cached and uncached lookups:

- ADABAS
- DATACOM

- DB2
- DL/1 BATCH (IMS)
- IDMS
- IMS ODBA
- KSDS
- SEQ

Configuring Lookups for IMS

You can use keys to retrieve data efficiently from IMS databases. PowerExchange enables you to build Concatenated Key (CCK) fields in the data map. Concatenated Key (CCK) fields enables PowerExchange to construct a fully-qualified Segment Search Argument (SSA) thereby improving IMS database search efficiency.

Fields specified in the Lookup condition transformation attribute are used by PowerExchange to create the Segment Search Argument (SSA). In order for a field to be used in the SSA, it must be marked as a key in the IMS source or target definition in Designer.

To provide search keys for IMS database lookups, use the following types of fields in the Lookup condition transformation attribute:

- **Concatenated Key (CCK) fields.** Specify these fields as keys in the IMS source or target definition and use them in the Lookup condition attribute. The use of CCK fields results in the most efficient search of the IMS database.
- **Key fields.** Specify these fields as key in the IMS source or target definition and use them in the Lookup condition attribute. You can specify either the CCK field or the key field for the desired segment as both will exist in the IMS source or target definition. If the segment is not the root, a combination of both CCK fields and key fields will likely be needed in the Lookup condition.
- **Search fields defined in the IMS Database Definition (DBD).** Specify these as keys in the IMS source or target definition and use them in the Lookup condition attribute. If the segment does not have a key, IMS can still scan the IMS segments by using an IMS search field. This type of search call is not as efficient as a keyed search with CCK fields or key fields. Assuming the root segment is keyed, include its CCK field with the search fields to limit the amount of data IMS scans and therefore the impact on the database.
- **Non-key or non-search fields.** The least efficient search method is to mark non-key fields or non-search fields as keys in the IMS source or target definition and in the Lookup condition attribute. This causes a scan of the IMS database to be done in order to find a match. This can adversely affect your IMS operational system and therefore should be avoided.

Tip: You can limit the amount of the database that will be scanned by specifying as many CCK and key fields as possible. If you use Search fields, include as many CCK fields as possible and, at minimum, the root CCK field. Only use non-key or non-search fields as a last resort.

For more information about creating CCK fields in IMS data maps, see the *PowerExchange Navigator User Guide*.

Configuring Lookups for CDC Data

Lookup transformations can be used in mappings extracting data from CDC sources including tables to which the CDC data is being applied.

If you use dynamic lookups for CDC data and sharing the lookup cache with other static lookups in the same mapping, specify both of the following custom properties in the session Custom Properties attribute field on the Config Object tab:

- MergeCDCReaders=Yes
- SingleThreadExecutionModel=Yes

These custom properties remove any partition points from the PWXPC CDC Reader through the transformations to the Writer. As a result, the order of the changes read from the change stream is maintained until the changes reach the Writer. This then ensures that any CDC data placed into a dynamic lookup cache is accessible to lookups sharing that cache in other pipelines.

Warning: The use of these custom properties will impact session throughput as it will single-thread all source data from the Reader through to the Writer. As a result, these custom properties should only be specified when there is a need to share CDC data stored in a dynamic cache across pipeline.

Stored Procedure Transformations

You can use Stored Procedure transformations that specify DB2 for z/OS or DB2 for i5/OS stored procedures in a PowerCenter mapping. Use Stored Procedure transformations for read or write operations in bulk data movement and change data capture (CDC) sessions.

Usage Considerations for Stored Procedure Transformations

The following usage considerations apply to Stored Procedure transformations for DB2 for z/OS and DB2 for i5/OS:

- You must create and configure an ODBC data source before you import the DB2 for z/OS or DB2 for i5/OS stored procedure.
- PowerCenter imports and calls stored procedures with unqualified procedure names that do not include a schema name. DB2 executes unqualified stored procedures for PowerExchange by using the CURRENT PATH register. DB2 searches the CURRENT PATH list of schema names from left to right until it finds a schema name for which a stored procedure definition exists.

However, by default, PowerExchange executes unqualified stored procedures by using the **User Name** value in the PWXPC connection as the schema name. For example, if the user name in the PowerExchange connection is USER1, PowerExchange calls unqualified stored procedures as USER1.*procedure_name*. Alternatively, for unqualified DB2 for i5/OS stored procedure names that are used in bulk data movement, you can use the **Library List** attribute in the PowerCenter DB2i5OS relational connection to qualify the stored procedure names.

Note: To explicitly set the schema name for a Stored Procedure transformation, edit the **Stored Procedure Name** property of the Stored Procedure transformation.

- If you use a Stored Procedure transformation with the **Resume from last checkpoint** recovery strategy, you must select the **Output is Deterministic** option for the transformation. If you do not select this option, you will not be able to validate the CDC session in Workflow Manager.
- After you configure the Stored Procedure transformation in a mapping, use the PowerCenter Integration Service to run the session.

Implementing a Stored Procedure Transformation in a Mapping

Implement a Stored Procedure transformation for a DB2 for z/OS or DB2 for i5/OS stored procedure in a PowerCenter mapping.

1. Create and compile the DB2 stored procedure on the z/OS or i5/OS system.
2. Verify that a PowerExchange Listener is running on the z/OS or i5/OS system.
3. On the Windows system where the Mapping Designer runs, create and configure an ODBC data source for the database that contains the stored procedure.

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

4. In the Mapping Designer, click **Transformation > Import Stored Procedure**.
5. In the **ODBC data source** box, enter or browse to the ODBC data source that you configured on the Windows system.
6. In the Mapping Designer, add the Stored Procedure transformation to the mapping.

Note: When you import the stored procedure, the Mapping Designer does not qualify the stored procedure name with the schema name. If you leave the name as is, PowerExchange executes any unqualified stored procedures by using the **User Name** value in the PowerExchange connection as the schema name. For unqualified DB2 for i5/OS stored procedure names that are used in PowerExchange bulk data movement, you can use the **Library List** attribute of the DB2i5OS relational connection to qualify the stored procedure name.

7. If the schema name that qualifies the stored procedure name is different from the user name under which the PowerExchange Listener runs, and if you did not use the **Library List** attribute to qualify DB2i5OS stored procedure names, edit the Stored Procedure transformation properties to specify the schema name. In the **Edit Transformations** dialog box, click the **Properties** tab and add the schema name before the stored procedure name in the **Stored Procedure Name** box.

For example, if the stored procedure name is ZEMPS1 and the user name under which the PowerExchange Listener runs is USER2, enter `USER2.ZEMPS` in the **Stored Procedure Name** box.

For more information about Stored Procedure transformations, see the *PowerCenter Transformation Guide*.

CHAPTER 4

Connections

This chapter includes the following topics:

- [Connections Overview, 58](#)
- [Configuring PowerExchange Connections, 61](#)
- [DB2 Relational Connections, 62](#)
- [DB2 CDC Application Connections, 70](#)
- [NRDB Batch Application Connections, 77](#)
- [NRDB CDC Application Connections, 82](#)
- [NRDB Lookup Relational Connections, 88](#)
- [Microsoft SQL Server Relational Connections, 91](#)
- [Microsoft SQL Server CDC Application Connections, 94](#)
- [Oracle Relational Connections, 99](#)
- [Oracle CDC Application Connections, 102](#)
- [Configuring Connection Attributes, 108](#)
- [Connection Attributes for Multiple-Record Writes, 126](#)
- [Asynchronous Network Communication, 127](#)

Connections Overview

Before the PowerCenter Integration Service can access a source or target in a session, you must configure connections in the Workflow Manager. When you create or modify a session that reads from or writes to a database or file, you can select only configured source and target databases. Connections are saved in the repository.

For PowerExchange Client for PowerCenter, you configure relational or application connections, depending on the source or the target type.

Connection Types for Extracting Source Data

The following table describes the type of connections you can create to extract data from PowerExchange, based on the data source and extraction mode:

Source Type and Extraction Mode	Connection	Connection Type
Adabas bulk	Application	PWX NRDB Batch
Adabas batch from condense files	Application	PWX NRDB CDC Change
Adabas continuous from PowerExchange Logger for LUW log files	Application	PWX NRDB CDC Real Time
Adabas real time	Application	PWX NRDB CDC Real Time
Datacom bulk	Application	PWX NRDB Batch
Datacom batch from condense files	Application	PWX NRDB CDC Change
Datacom continuous from PowerExchange Logger for LUW log files	Application	PWX NRDB CDC Real Time
Datacom real time	Application	PWX NRDB CDC Real Time
DB2 for i5/OS bulk	Relational	PWX DB2i5OS
DB2 for i5/OS batch from condense files	Application	PWX DB2i5OS CDC Change
DB2 for i5/OS continuous from PowerExchange Logger for LUW log files	Application	PWX DB2i5OS CDC Real Time
DB2 for i5/OS real time	Application	PWX DB2i5OS CDC Real Time
DB2 for Linux, UNIX, and Windows bulk	Relational	PWX DB2LUW
DB2 for Linux, UNIX, and Windows batch from PowerExchange Logger for LUW log files	Application	PWX DB2LUW CDC Change
DB2 for Linux, UNIX, and Windows continuous from PowerExchange Logger for LUW log files	Application	PWX DB2LUW CDC Real Time
DB2 for Linux, UNIX, and Windows real time	Application	PWX DB2LUW CDC Real Time
DB2 for z/OS bulk	Relational	PWX DB2zOS
DB2 for z/OS batch from condense files	Application	PWX DB2zOS CDC Change
DB2 for z/OS continuous from PowerExchange Logger for LUW log files	Application	PWX DB2zOS CDC Real Time
DB2 for z/OS real time	Application	PWX DB2zOS CDC Real Time
IDMS bulk	Application	PWX NRDB Batch
IDMS batch from condense files	Application	PWX NRDB CDC Change
IDMS continuous from PowerExchange Logger for LUW log files	Application	PWX NRDB CDC Real Time

Source Type and Extraction Mode	Connection	Connection Type
IDMS real time	Application	PWX NRDB CDC Real Time
IMS bulk	Application	PWX NRDB Batch
IMS batch from condense files	Application	PWX NRDB CDC Change
IMS continuous from PowerExchange Logger for LUW log files	Application	PWX NRDB CDC Real Time
IMS real time	Application	PWX NRDB CDC Real Time
Microsoft SQL Server bulk	Relational	PWX MSSQLServer
Microsoft SQL Server batch from PowerExchange Logger for LUW log files	Application	PWX MSSQL CDC Change
Microsoft SQL Server continuous from PowerExchange Logger for LUW log files	Application	PWX MSSQL CDC Real Time
Microsoft SQL Server real time	Application	PWX MSSQL CDC Real Time
Oracle bulk	Relational	PWX Oracle
Oracle batch from PowerExchange Logger for LUW log files	Application	PWX Oracle CDC Change
Oracle continuous from PowerExchange Logger for LUW log files	Application	PWX Oracle CDC Real Time
Oracle real time	Application	PWX Oracle CDC Real Time
Sequential and flat files bulk	Application	PWX NRDB Batch
Sequential and flat files batch from condense files	n/a	n/a
Sequential and flat files real time	n/a	n/a
VSAM bulk	Application	PWX NRDB Batch
VSAM batch from condense files	Application	PWX NRDB CDC Change
VSAM continuous from PowerExchange Logger for LUW log files	Application	PWX NRDB CDC Real Time
VSAM real time	Application	PWX NRDB CDC Real Time

Connection Types for Loading Target Data

The following table describes the types of relational and application connections that you can define to load data to PowerExchange targets:

Target Type	Connection	Connection Type
Adabas bulk	Application	PWX NRDB Batch
DB2 for i5/OS bulk	Relational	PWX DB2i5OS

Target Type	Connection	Connection Type
DB2 for Linux, UNIX, and Windows bulk	Relational	PWX DB2LUW
DB2 for z/OS bulk	Relational	PWX DB2zOS
IMS bulk	Application	PWX NRDB Batch
Microsoft SQL Server bulk	Relational	PWX MSSQLServer
Oracle bulk	Relational	PWX Oracle
Sequential and flat files bulk	Application	PWX NRDB Batch
VSAM bulk	Application	PWX NRDB Batch

Configuring PowerExchange Connections

Configure PWXPC connections to data sources and targets. Use the connections in sessions that extract bulk data or change data from a relational or nonrelational source or that load bulk data to a relational or nonrelational target.

1. In the Workflow Manager, connect to a PowerCenter repository.
2. To configure a PWXPC relational connection, click **Connections > Relational**.
The **Relational Connection Browser** dialog box appears.
3. To configure a PWXPC application connection, click **Connections > Application**.
The **Application Connection Browser** dialog box appears.
4. In the **Select Type** field, select the specific type of connection.
5. Click **New**.
The **Connection Object Definition** dialog box appears.
6. Enter values for the connection attributes.
For more information, see the topic for the specific connection type.
7. Click **OK**.
The new connection appears in the Application or Relational Object Browser.

RELATED TOPICS:

- [“DB2 CDC Application Connections” on page 70](#)
- [“DB2 Relational Connections” on page 62](#)
- [“Microsoft SQL Server CDC Application Connections” on page 94](#)
- [“Microsoft SQL Server Relational Connections” on page 91](#)
- [“NRDB Batch Application Connections” on page 77](#)
- [“NRDB CDC Application Connections” on page 82](#)
- [“NRDB Lookup Relational Connections” on page 88](#)

- [“Oracle CDC Application Connections” on page 102](#)
- [“Oracle Relational Connections” on page 99](#)

DB2 Relational Connections

Configure relational connections to extract data from and load data to DB2 for z/OS, DB2 for i5/OS, and DB2 for Linux, UNIX, and Windows sources and targets in bulk data movement sessions. Also use relational connections to apply change data to DB2 relational targets in CDC sessions.

The following table describes the connection attributes for the PWX DB2zOS, PWX DB2i5OS, and PWX DB2LUW relational connection types:

Connection Attribute	Required	DB2zOS, DB2i5OS, DB2LUW Types	Description
Name	Yes	All	The name of the relational connection.
Code Page	Yes	All	The code page that the PowerCenter Integration Service uses to extract data from the source database. Note: In Unicode mode, PWXPC sets the code page with this value overriding any code page specification in the PowerExchange dbmover configuration file.
Location	Yes	All	The location of the source or target database as specified as specified in a NODE statement in the PowerExchange dbmover configuration file.
Database Name	Yes	All	The DB2 subsystem or database name.
User Name	Yes	All	A user name that can be used to access the database. For databases on supported Linux, UNIX, or Windows systems, if you have enabled PowerExchange LDAP user authentication and disabled relational pass-through authentication, the user name is the enterprise user name. For more information, see the <i>PowerExchange Reference Manual</i> .

Connection Attribute	Required	DB2zOS, DB2i5OS, DB2LUW Types	Description
Password	Yes	All	<p>A password for the specified user. Passwords that are less than nine characters in length are not validated.</p> <p>To avoid errors that might arise due to code page differences, Informatica recommends that the password includes only those characters that are allowed for passphrases.</p> <p>For DB2i5OS and DB2zOS relational connections, you can enter a valid PowerExchange passphrase instead of a password. An i5/OS passphrase can be from 9 to 31 characters in length when you use a PWXPC connection. A z/OS passphrase can be from 9 to 128 characters in length. A passphrase can contain the following characters:</p> <ul style="list-style-type: none"> - Uppercase and lowercase letters - The numbers 0 to 9 - Spaces - The following special characters: ' - ; # \ , . / ! % & * () _ + { } : @ < > ? <p>Note: The first character is an apostrophe.</p> <p>Passphrases cannot include single quotation marks ('), double quotation marks ("), or currency symbols.</p> <p>On z/OS, the allowable characters in the IBM IRRPHREX exit do not affect the allowable characters in PowerExchange passphrases.</p> <p>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.</p> <p>To use passphrases, ensure that the PowerExchange Listener runs with a security setting of SECURITY=(1,N) or higher in the DBMOVER member. For more information, see "SECURITY Statement" in the <i>PowerExchange Reference Manual</i>.</p>
Isolation Level	Yes	DB2i5OS	<p>The commit scope of the transaction.</p> <p>Valid values:</p> <ul style="list-style-type: none"> - None - CS. Cursor stability. - RR. Repeatable Read. - CHG. Change. - ALL <p>Default is CS.</p>

Connection Attribute	Required	DB2zOS, DB2i5OS, DB2LUW Types	Description
Database file overrides	No	DB2i5OS	<p>Override for the i5/OS database file. The format is:</p> <pre>from_file/to_library/to_file/to_member</pre> <p>Where:</p> <ul style="list-style-type: none"> - <i>from_file</i> is the file to be overridden. - <i>to_library</i> is the new library to use. - <i>to_file</i> is the file in the new library to use. - <i>to_member</i> is optional and is the member in the new library and file to use. If no value is specified, *FIRST is used. <p>You can enter up to eight unique file overrides on a single connection. A single override applies to a single source or target. When you specify multiple file overrides, include a space between each file override and enclose the entire string of file overrides in double quotation marks.</p> <p>Note: If both Library List and Database file overrides are specified and a table exists in both, Database file overrides takes precedence.</p>
Library List	No	DB2i5OS	<p>A list of the libraries that PowerExchange searches to qualify a table name for Select, Insert, Delete, or Update statements or to qualify a stored procedure name when the procedure is called for execution. PowerExchange searches the list if a table name or stored procedure name is not qualified with a schema name.</p> <p>If you specify multiple libraries, separate the library names with spaces and enclose the list in double quotation marks, for example, "TGTLIB1 TGTLIB2 TGTLIB3".</p> <p>Note: If both Library List and Database file overrides are specified and a table exists in both, Database file overrides takes precedence.</p>
Environment SQL	No	All	SQL commands that run in the database environment.
Compression	No	All	Select this option to compress source data during the PowerCenter session.
Encryption Type	Yes	All	<p>The encryption type. For information about supported encryption types, see "Configuring Encryption and Compression" on page 109.</p> <p>Default is None.</p>
Encryption Level	No	All	<p>The encryption level. For information about supported encryption levels, see "Configuring Encryption and Compression" on page 109.</p> <p>Default is 1.</p>
Pacing Size	Yes	All	<p>The pacing size. For information about setting the pacing size, see "Configuring Pacing" on page 110.</p> <p>Default is 0.</p>
Interpret as Rows	No	All	Controls whether the pacing size is in number of rows.

Connection Attribute	Required	DB2zOS, DB2i5OS, DB2LUW Types	Description
Bulk Load	No	DB2zOS	Controls whether PowerExchange loads data to DB2 for z/OS targets with the DB2 LOAD utility. If you select this option, you can configure the remaining connection attributes that apply to the DB2 LOAD utility. Otherwise, PowerExchange ignores these attributes.
Filename	No	DB2zOS	The data set prefix that PowerExchange uses to create the temporary files that the DB2 LOAD utility uses to load data to a DB2 table.
Space	Yes	DB2zOS	Controls whether to allocate z/OS space in tracks or cylinders. Valid values: - TRACK - CYLINDER Default is TRACK.
Primary Space	Yes	DB2zOS	The primary space on z/OS to use for LOAD operations. Default is 0.
Secondary Space	Yes	DB2zOS	The secondary space on z/OS to use for LOAD operations. Default is 0.
Delete Temporary Files	Yes	DB2zOS	Determines how PowerExchange handles the temporary files it creates for the DB2 LOAD utility to load data into a DB2 table. Valid values: - NO . Do not delete the temporary files. - BEFORE . Delete the temporary files before running the utility. - AFTER SUCCESS ONLY . Delete the temporary files if the utility ends successfully with return code 0. - AFTER . Delete the temporary files after the utility runs. Default is NO.
JCL Template	Yes	DB2zOS	The name of the JCL template for the DB2 LOAD utility on the PowerExchange target system. Default is DB2LDJCL.
CTL Template	Yes	DB2zOS	The name of the control file template for the DB2 LOAD utility on the PowerExchange on the target system. Default is DB2LDCTL.
Load Options	Yes	DB2zOS	Indicates how the data that PowerExchange provides to the DB2 LOAD utility is loaded into a DB2 table. Valid values: - RESUME . Generate a LOAD RESUME statement. - REPLACE . Generate a LOAD REPLACE statement. Default is RESUME.

Connection Attribute	Required	DB2zOS, DB2i5OS, DB2LUW Types	Description
Mode Type	Yes	DB2zOS	<p>Indicates how PowerExchange runs the DB2 LOAD utility to load data into a DB2 table.</p> <p>Valid values:</p> <ul style="list-style-type: none"> - TASK. Run the LOAD utility as a subtask under the PowerExchange Listener. - JOB. Submit a separate job to run the DB2 LOAD utility. - NOSUBMIT. Create the files and JCL to run the DB2 LOAD utility (unless Mode Time is set to DATAONLY) but do not submit the load job. You must submit the job manually. <p>Default is TASK.</p>
Mode Time	Yes	DB2zOS	<p>Determines how PowerExchange handles the execution of the DB2 LOAD utility.</p> <p>Valid values:</p> <ul style="list-style-type: none"> - WAIT. Wait for the job to end before returning control to PowerCenter. This option can be specified only when the Mode Type is JOB or TASK. - NO WAIT. Return to PowerCenter without waiting for the job to end. This option can be specified only when Mode Type is set to JOB or NOSUBMIT. - TIMED. Wait the number of seconds that is specified in the Time attribute before returning control to PowerCenter. The TIMED option can be specified only when Mode Type is set to JOB. - DATAONLY. Create the data file only. Do not create the files and JCL for running the DB2 LOAD utility. Usually, this option is used when Mode Type is set to NOSUBMIT. <p>Default is WAIT.</p> <p>Notes:</p> <ul style="list-style-type: none"> - If you enter WAIT, PowerExchange uses 99,999 seconds as the network operation timeout value instead of the value specified by the TCP/IP_OP_TIMEOUT parameter of the PWX Override connection attribute. If you enter TIMED, PowerExchange adds 5 minutes to the network operation timeout value specified by the connection attribute. - If you enter WAIT and the submitted job fails when you run the workflow, the PowerExchange Listener on the z/OS system continues to run. To stop the Listener, perform one of the following actions: <ul style="list-style-type: none"> - Enter the following command from the z/OS operator console: <pre>F task_name,STOPTASK TASKID=task_id</pre> <p>For more information, see the "STOPTASK Command" topic in the <i>PowerExchange Command Reference</i>.</p> - Enter the STOPTASK command in the Database Row Test dialog box of the PowerExchange Navigator. Select TASK_CNTL in the DB Type list, and select Stop Task in the Fetch box. The SQL Statement box displays <code>stoptask taskid=</code>. Enter a task ID. <p>For more information, see the topic, "Issuing PowerExchange Listener Commands in a Database Row Test," in the <i>PowerExchange Navigator User Guide</i>.</p>

Connection Attribute	Required	DB2zOS, DB2i5OS, DB2LUW Types	Description
Time	Yes	DB2zOS	The wait time, in seconds, that is in effect when you set Mode Type to JOB and Mode Time to TIMED . Valid values are 1 to 99998. Default is 0.
Convert character data to string	No	All	Controls whether character fields are converted to string fields so that embedded nulls in data are processed as spaces. By default, this attribute is not selected. For more information, see "Converting Character Data to Strings" on page 110 .
Write Mode	No	All	The write mode. For information about Write Mode options, see "Configuring Write Mode" on page 112 . Default is Confirm Write On .
Reject File	No	All	Overrides the default prefix of PWXR for the reject file. PowerExchange creates the reject file on the target machine when the Write Mode is Asynchronous with Fault Tolerance . Note: Enter PWXDISABLE to prevent creation of the reject files.
Correlation Id	No	DB2zOS	A value to use as the DB2 Correlation ID for DB2 requests. This value overrides the value in the SESSID statement in the PowerExchange DBMOVER configuration file.
Offload Processing	No	DB2zOS	Indicates whether to use offload processing to move PowerExchange bulk data processing from the source system to the PowerCenter Integration Service machine. Valid values: <ul style="list-style-type: none"> - No. Do not use offload processing. - Yes. Use offload processing. - Auto. PowerExchange determines whether to use offload processing. Default is No.
Worker Threads	No	DB2zOS	When offload processing is enabled, specifies the number of threads that PowerExchange uses on the PowerCenter Integration Service machine to process bulk data. For optimal performance, this value should not exceed the number of installed or available processors on the PowerCenter Integration Service machine. Valid values are 1 through 64. Default is 0, which disables multithreading.

Connection Attribute	Required	DB2zOS, DB2i5OS, DB2LUW Types	Description
Array Size	No	DB2zOS, DB2i5OS	<p>The DB2 fetch array size, in number of rows, for DB2 bulk data movement operations that use the DB2 access method. The array size pertains to the DB2 multiple-row FETCH statements that PowerExchange uses to read data from DB2 source tables.</p> <p>Valid values are from 1 through 5000. Default is 25.</p> <p>Tip: For DB2 for i5/OS bulk data movement operations, Informatica recommends that you enter an array size no greater than 100.</p> <p>Note: PowerExchange dynamically lowers the array size when all the following conditions are true:</p> <ul style="list-style-type: none"> - The table contains LOB columns. - The Array Size value is greater than 1. - Row size * Array Size is greater than 16000000 bytes.

Connection Attribute	Required	DB2zOS, DB2i5OS, DB2LUW Types	Description
PWX Override	No	All	<p>PowerExchange connection overrides, separated by a semicolon. You can enter one or more of the following optional overrides:</p> <ul style="list-style-type: none"> - APPBUFSIZE=<i>app_buf_size</i>. Specifies the initial size of the application buffer for the connection. Overrides the APPBUFSIZE statement in the DBMOVER configuration file. - CNV_ERROR_OK= {N Y}. If you use a DB2 relational connection to write data to a DB2 target, set this parameter to Y to allow the workflow continue after a PWX-00143 message reports a DBAPI illegal null conversion error. This error occurs when the workflow tries to write null data to a target NOT NULL column. Set this parameter to N to have the workflow end abnormally when this type of error occurs. Default is N. - CONNECTSTRINGCODEPAGE=<i>code_page</i>. Code page of the characters in the connection string. Enter this override if PowerExchange issues message PWX-07610. - CONNSHARE={N Y}. By default, all DB2 lookups in a workflow use the same connection, and the PowerExchange Listener performs them in a single task. To preserve the behavior in effect for DB2 lookups in PowerExchange releases earlier than 9.6.0, specify CONNSHARE=N. Caution: If the DB2 connection is used as a target in a CDC workflow, do not change the default behavior. Otherwise, internal PowerCenter state tables that require connection sharing might not be updated correctly. Note: DB2 lookups that share a connection do not use offload processing, partitioning, or threading. - DB2TRUNCASDEL={N Y}. Applies to DB2zOS target connections only. With the default of N, when the PowerCenter Integration Service requests a TRUNCATE statement, PowerExchange issues a TRUNCATE statement. If you set this override to Y, when the PowerCenter Integration Service requests a TRUNCATE statement, PowerExchange issues a DELETE statement. Note: The DB2TRUNCASDEL override applies only to DB2zOS target connections. For DB2i5OS target connections, PowerExchange always issues a DELETE statement. For DB2LUW target connections, PowerExchange always issues a TRUNCATE statement. - LOWVALUES={Y N}. Indicates whether PowerExchange preserves hexadecimal '0' values, called <i>low values</i>, in source character fields when passing these values to a PowerCenter session. Enter Y to preserve low values. When the session runs, PWXPC can write these values to a VSAM target on z/OS or to a sequential file target on Linux, UNIX, Windows, or z/OS. Overrides the LOWVALUES statement in the DBMOVER configuration file. - PWXNOQUOTES={Y N}. By default, in the SQL SELECT statements that PWXPC uses to select data for session processing, PWXPC adds double-quotation marks (") around every table name and field name that includes spaces or special characters. These special characters are /+ -= ~ ` ! % ^ & * () [] { } ' ; ? , < > \ \ \ . If a table is identified by both a schema name and table name in the format <i>schema.table_name</i>, PWXPC places the quotation marks only around the "table_name" value.

Connection Attribute	Required	DB2zOS, DB2i5OS, DB2LUW Types	Description
			<p>If for any reason, you cannot use quoted table names or field names in your environment, set PWXNOQUOTES to Y to override the default behavior.</p> <ul style="list-style-type: none"> - RETLOGINFOMSG={N Y}. By default, PWXPC writes PowerExchange error and warning messages but not informational messages to the session log. If you specify RETLOGINFOMSG=Y, PWXPC writes informational messages as well as error and warning messages to the session log. - TCPIP_OP_TIMEOUT=<i>network_operation_timeout</i>. A network operation timeout interval in seconds. For more information about this timeout override, see “Asynchronous Network Communication” on page 127. - TCPIP_CON_TIMEOUT=<i>connection_timeout</i>. A connection timeout interval in seconds. For more information about this timeout override, see “Asynchronous Network Communication” on page 127.
Connection Retry Period	No	All	<p>The number of seconds the PowerCenter Integration Service attempts to reconnect to the PowerExchange Listener after the initial connection attempt fails. If the Integration Service cannot connect to the PowerExchange Listener within the retry period, the session fails.</p> <p>Default value is 0, which disables connection retries based on this PWXPC connection attribute.</p> <p>Note: PowerExchange Listener connections to source databases do not have connection resilience.</p>

RELATED TOPICS:

- [“Configuring Encryption and Compression” on page 109](#)
- [“Configuring Pacing” on page 110](#)
- [“Converting Character Data to Strings” on page 110](#)
- [“Configuring Write Mode” on page 112](#)

DB2 CDC Application Connections

Configure CDC application connections to extract change data from DB2 for z/OS, DB2 for i5/OS, or DB2 for Linux, UNIX, and Windows sources in real-time extraction mode. Also configure CDC application connections to extract change data from PowerExchange Condense files in batch extraction mode or from PowerExchange Logger for Linux, UNIX, and Windows log files in batch or continuous extraction mode.

Depending on the DB2 source type, you can configure the following types of application connections:

- PWX DB2zOS CDC Change
- PWX DB2zOS CDC Real Time
- PWX DB2i5OS CDC Change
- PWX DB2i5OS CDC Real Time
- PWX DB2LUW CDC Change

- PWX DB2LUW CDC Real Time

The following table describes the connection attributes:

Connection Attribute	Required?	Change or Real Time	Description
Name	Yes	Both	The name of the application connection.
Code Page	Yes	Both	<p>The code page that the PowerCenter Integration Service uses to extract data from the source database.</p> <p>In Unicode mode, you must select UTF-8 for DB2LUW CDC connections.</p> <p>Note: In Unicode mode, PWXPC sets the code page with this value overriding any code page specification in the PowerExchange configuration file.</p>
Location	Yes	Both	The location of the source or target database as specified in a NODE statement in the PowerExchange dbmover.cfg configuration file.
User Name	Yes	Both	<p>A user name that can be used to access the DB2 database.</p> <p>For a database on a supported Linux, UNIX, or Windows system, if you have enabled PowerExchange LDAP user authentication, the user name is the enterprise user name. For more information, see the <i>PowerExchange Reference Manual</i>.</p>
Password	Yes	Both	<p>A password for the specified user. Passwords that are less than nine characters in length are not validated.</p> <p>To avoid errors that might arise due to code page differences, Informatica recommends that the password includes only those characters that are allowed for passphrases.</p> <p>For PWX DB2i5OS and DB2zOS CDC, you can enter a valid PowerExchange passphrase instead of a password. 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 when you use a PWXPC connection. A passphrase can contain the following characters:</p> <ul style="list-style-type: none"> - Uppercase and lowercase letters - The numbers 0 to 9 - Spaces - The following special characters: ' - ; # \ , . / ! % & * () _ + { } : @ < > ? <p>Note: The first character is an apostrophe.</p> <p>Passphrases cannot include single quotation marks ('), double quotation marks ("), or currency symbols.</p> <p>On z/OS, the allowable characters in the IBM IRRPHREX exit do not affect the allowable characters in PowerExchange passphrases.</p> <p>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.</p> <p>To use passphrases, ensure that the PowerExchange Listener runs with a security setting of SECURITY=(1,N) or higher in the DBMOVER member. For more information, see "SECURITY Statement" in the <i>PowerExchange Reference Manual</i>.</p>
Compression	No	Both	Select this option to compress source data during the PowerCenter session.

Connection Attribute	Required?	Change or Real Time	Description
Encryption Type	Yes	Both	The encryption type. For information about encryption types, see “Configuring Encryption and Compression” on page 109 . Default is None.
Encryption Level	No	Both	The encryption level. For information about encryption levels, see “Configuring Encryption and Compression” on page 109 . Default is 1.
Pacing Size	Yes	Both	The pacing size. For information about setting the pacing size, see “Configuring Pacing” on page 110 . Default is 0.
Interpret as Rows	No	Both	Controls whether the pacing size is in number of rows.
Image Type	No	Both	Indicates whether PWXPC passes captured Update operations to CDC sessions as Updates, or as Deletes followed by Inserts. Default is BA, which processes Updates as Deletes followed by Inserts. For more information, see “Configuring Image Type” on page 113 .
Application Name	No	Both	The application name.
RestartToken File Folder	Yes	Both	The restart token folder. Default is \$PMRootDir/Restart. For more information, see “Configuring CDC Restart Attributes” on page 117 .
RestartToken File Name	No	Both	The restart token file name.
Number of Runs to Keep RestartToken File	No	Both	The maximum number of backup copies of the restart token file to keep. Default is 0.
Recovery Cache Folder	No	Both	The file cache folder to enable recovery for the session. Default is \$PMRootDir/Cache.
UOW Count	No	Both	The number of units of work (UOWs) that PWXPC processes before it sends a commit request to PowerCenter to commit data to all targets in the CDC session. Valid values: -1. Disables this attribute. 0. Disables this attribute. n. Specifies the number of UOWs that PWXPC processes before sending a commit request. Default is 1. For more information, see “Configuring UOW Count” on page 118 .

Connection Attribute	Required?	Change or Real Time	Description
Reader Time Limit	No	Real Time	<p>The number of seconds that the PowerCenter Integration Service reads data from the source before stopping.</p> <p>If you enter 0, the Reader Time Limit attribute does not limit the reader time. This attribute is intended for testing purposes only.</p> <p>Tip: Use Idle Time instead of Reader Time Limit.</p> <p>Default is 0.</p>
Idle Time	No	Real Time	<p>The number of seconds the PowerExchange Listener remains idle after reaching the end of the change log, as indicated by message PWX-09967, before returning an end-of-file (EOF).</p> <p>Valid values:</p> <ul style="list-style-type: none"> -1. The EOF is not returned. The session runs continuously. 0. The EOF is returned at the end of log. The session ends successfully. <i>n</i>. The number of seconds after which the EOF is returned. <p>Default is -1.</p> <p>For more information, see "Configuring Idle Time" on page 116.</p>
Real-Time Flush Latency in milli-seconds	No	Real Time	<p>The number of milliseconds that must elapse before PWXPC sends a commit request to PowerCenter. When this period elapses, PWXPC continues to read the changes in the current UOW until it reaches the end of the UOW. Then PWXPC sends a commit request to PowerCenter so that the data can be committed to the targets.</p> <p>Valid values:</p> <ul style="list-style-type: none"> -1. Disables this attribute. 0 to 86400. Specifies the number of milliseconds that must elapse before PWXPC sends a commit request to PowerCenter. If you specify a value from 0 through 2000, PWXPC uses 2000 milliseconds (2 seconds). <p>Default is 0.</p> <p>For more information, see "Configuring PWX Latency and Real-Time Flush Latency" on page 119.</p>
PWX Latency in seconds	No	Real Time	<p>The maximum number of seconds that PowerExchange extraction processing on the source platform waits for more change data before it returns control to PWXPC on the PowerCenter Integration Service machine.</p> <p>Default is 2.</p>
Maximum Rows Per commit	No	Both	<p>The maximum number of change records that PWXPC processes before it sends a commit request to PowerCenter to commit data to all targets in the CDC session. PWXPC does not wait for a UOW boundary to commit the change data.</p> <p>Default is 0, which disables this attribute.</p> <p>For more information, see "Configuring Maximum Rows per Commit" on page 120.</p>

Connection Attribute	Required?	Change or Real Time	Description
Minimum Rows Per commit	No	Real Time	<p>The minimum number of change records that the PowerExchange Listener must read from the change stream before it passes a commit record to PWXPC. Before reaching this minimum, the PowerExchange Listener passes only change records to PWXPC and discards any commit records.</p> <p>Default is 0, which disables this attribute.</p> <p>For more information, see "Configuring Minimum Rows per Commit" on page 121.</p>
Journal Name Note: For PWX DB2i50S CDC Real Time application connections only.	No	Both	<p>Overrides the library and journal name in the PowerExchange CAPL_CONNECTION.</p> <p>Specify complete library and journal names in the format:</p> <p><i>library/journal</i></p>
Library/File Override Note: For PWX DB2i50S CDC Real Time application connections only.	No	Both	<p>Overrides the library and file name in the extraction maps of all sources that use the connection.</p> <p>Specify a single library and file name in the format:</p> <p><i>library/file</i></p> <p>If specifying asterisk for the library name, use the session properties Library/File Override attribute.</p> <p>Note: This attribute is overridden by the Library/File Override value on the session properties.</p>
Convert character data to string	No	Both	<p>Controls whether to convert character fields to string fields so that embedded nulls in data are processed as spaces.</p> <p>By default, this attribute is not selected.</p> <p>For more information, see "Converting Character Data to Strings" on page 110.</p>
Event Table	No	Real Time	<p>The PowerExchange extraction map name that is used for event processing.</p> <p>For more information, see "Configuring Event Table Processing" on page 114.</p>
Offload Processing	No	Real Time	<p>Controls whether PowerExchange uses CDC offload processing. When offload processing is enabled, PowerExchange transfers column-level processing of the change data from the source system to the PowerCenter Integration Service machine.</p> <p>Valid values:</p> <ul style="list-style-type: none"> - No. Disables offload processing. - Yes. Enables offload processing. - Auto. PowerExchange determines whether to use offload processing. <p>Default is No.</p> <p>For more information about offload processing, see the <i>PowerExchange CDC Guide for Linux, UNIX, and Windows</i>.</p>

Connection Attribute	Required?	Change or Real Time	Description
Worker Threads	No	Real Time	<p>If you use CDC offload processing, the number of threads that PowerExchange uses on the PowerCenter Integration Service machine to process change data.</p> <p>For optimal performance, this value should not exceed the number of installed or available processors on the Integration Service machine.</p> <p>Valid values are from 1 through 64.</p> <p>Default is 0.</p>
Array Size	No	Real Time	<p>If the Worker Threads value is greater than 0, specifies the size of the storage array, in number of records, for the threads.</p> <p>Valid values are from 1 through 5000.</p> <p>Default is 25.</p>
Map Location	No	Real Time	<p>If you use remote logging to a PowerExchange Logger for Linux, UNIX, and Windows, the location of the source extraction maps. This value must be the node name of the location, as specified in a NODE statement in the PowerExchange dbmover.cfg configuration file on the Integration Service machine.</p> <p>For more information about remote logging, see the <i>PowerExchange CDC Guide for Linux, UNIX, and Windows</i>.</p>
Map Location User	No	Real Time	If you specify a Map Location value, a valid user ID for the map location.
Map Location Password	No	Real Time	If you specify a Map Location value, a valid password for the specified map location user.
CAPI Connection Name Override	No	Real Time	<p>Overrides the default CAPI connection name.</p> <p>For more information, see “Configuring the CAPI Connection Name Override” on page 115.</p>
Retrieve PWX Log Entries	No	Both	<p>Controls whether PWXPC writes PowerExchange informational messages related to a CDC session in the PowerCenter session log.</p> <p>Default is to write PowerExchange error and warning messages but not informational messages to the session log.</p>

Connection Attribute	Required?	Change or Real Time	Description
PWX Override	No	Both	<p>PowerExchange connection overrides, separated by a semicolon. You can enter one or more of the following optional overrides:</p> <ul style="list-style-type: none"> - APPBUFSIZE=<i>app_buf_size</i>. Specifies the initial size of the application buffer for the connection. Overrides the APPBUFSIZE statement in the DBMOVER configuration file. - CONNECTSTRINGCODEPAGE=<i>code_page</i>. Code page of the characters in the connection string. Enter this override if PowerExchange issues message PWX-07610. - LOWVALUES={Y N}. Indicates whether PowerExchange preserves hexadecimal '0' values, called <i>low values</i>, in source character fields when passing these values to a PowerCenter session. Enter Y to preserve low values. When the session runs, PWXPC can write these values to a VSAM target on z/OS or to a sequential file target on Linux, UNIX, Windows, or z/OS. Overrides the LOWVALUES statement in the DBMOVER configuration file. - PWXNOQUOTES={Y N}. By default, in the SQL SELECT statements that PWXPC uses to select data for session processing, PWXPC adds double-quotation marks (") around every table name and field name that includes spaces or special characters. These special characters are /+~=^~!%&*() [] {} ' ; ? , < > \ \ . If a table is identified by both a schema name and table name in the format <i>schema.table_name</i>, PWXPC places the quotation marks only around the "<i>table_name</i>" value. If for any reason, you cannot use quoted table names or field names in your environment, set PWXNOQUOTES to Y to override the default behavior. - RETLOGINFOMSG={N Y}. By default, PWXPC writes PowerExchange error and warning messages but not informational messages to the session log. If you specify RETLOGINFOMSG=Y, PWXPC writes informational messages as well as error and warning messages to the session log. Note: The RETLOGINFOMSG connection override is equivalent to the Retrieve PWX Log Entries connection attribute. - TCPIP_OP_TIMEOUT=<i>network_operation_timeout</i>. A network operation timeout interval in seconds. For more information about this timeout override, see "Asynchronous Network Communication" on page 127. - TCPIP_CON_TIMEOUT=<i>connection_timeout</i>. A connection timeout interval in seconds. For more information about this timeout override, see "Asynchronous Network Communication" on page 127.
Connection Retry Period	No	Both	<p>The number of seconds the PowerCenter Integration Service attempts to reconnect to the PowerExchange Listener after the initial connection attempt fails. If the Integration Service cannot connect to the PowerExchange Listener within the retry period, the session fails.</p> <p>Default value is 0, which disables connection retries based on this PWXPC connection attribute.</p> <p>Note: PowerExchange Listener connections to source databases do not have connection resilience.</p>

RELATED TOPICS:

- ["Configuring Encryption and Compression" on page 109](#)
- ["Configuring Pacing" on page 110](#)

- [“Converting Character Data to Strings” on page 110](#)
- [“Retrieving PowerExchange Log Entries” on page 111](#)
- [“Configuring Image Type” on page 113](#)
- [“Configuring the CAPI Connection Name Override” on page 115](#)
- [“Configuring Idle Time” on page 116](#)
- [“Configuring CDC Restart Attributes” on page 117](#)
- [“Configuring UOW Count” on page 118](#)
- [“Configuring PWX Latency and Real-Time Flush Latency” on page 119](#)
- [“Configuring Maximum Rows per Commit” on page 120](#)
- [“Commit Processing for CDC Sessions” on page 157](#)
- [“Managing Session Log and Restart Token File History” on page 181](#)
- [“Configuring Event Table Processing” on page 114](#)
- [“Enabling Recovery for a CDC Session” on page 173](#)

NRDB Batch Application Connections

Configure NRDB Batch application connections to extract data from and load data to nonrelational sources and targets in bulk data movement sessions. Also use NRDB Batch application connections to apply change data to nonrelational targets in CDC sessions.

The following table describes the connection attributes for the PWX NRDB Batch application connection type:

Connection Attribute	Required	Description
Name	Yes	The name of the application connection.
Code Page	Yes	The code page for the PowerCenter Integration Service to use to extract data from the source database. Note: In Unicode mode, PWXPC sets the code page with this value overriding any code page specification in the PowerExchange configuration file.
Location	Yes	The location of the source or target database as specified in a NODE statement in the PowerExchange dbmover.cfg configuration file.
User Name	Yes	The database user name. For databases on supported Linux, UNIX, or Windows systems, if you have enabled PowerExchange LDAP user authentication, the user name is the enterprise user name. For more information, see the <i>PowerExchange Reference Manual</i> .

Connection Attribute	Required	Description
Password	Yes	<p>A password for the specified user or a valid PowerExchange passphrase. Passwords that are less than nine characters in length are not validated.</p> <p>To avoid errors that might arise due to code page differences, Informatica recommends that the password includes only those characters that are allowed for passphrases.</p> <p>For nonrelational sources or targets on z/OS, and for sequential source or target files on i5/OS, you can enter a passphrase instead of a password. 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 with a PWXPC connection. A passphrase can contain the following characters:</p> <ul style="list-style-type: none"> - Uppercase and lowercase letters - The numbers 0 to 9 - Spaces - The following special characters: ' - ; # \ , . / ! % & * () _ + { } : @ < > ? <p>Note: The first character is an apostrophe.</p> <p>Passphrases cannot include single quotation marks ('), double quotation marks ("), or currency symbols.</p> <p>On z/OS, the allowable characters in the IBM IRRPHREX exit do not affect the allowable characters in PowerExchange passphrases.</p> <p>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.</p> <p>To use passphrases, ensure that the PowerExchange Listener runs with a security setting of SECURITY=(1,N) or higher in the DBMOVER member. For more information, see "SECURITY Statement" in the <i>PowerExchange Reference Manual</i>.</p> <p>To use passphrases for IMS connections, ensure that the following additional requirements are met:</p> <ul style="list-style-type: none"> - You must configure ODBA access to IMS as described in the <i>PowerExchange Navigator User Guide</i>. - 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.
Compression	No	Controls whether to compress source data during the session.
Encryption Type	No	<p>The encryption type. For information about encryption types, see "Configuring Encryption and Compression" on page 109.</p> <p>Default is None.</p>
Encryption Level	No	<p>The encryption level. For information about encryption levels, see "Configuring Encryption and Compression" on page 109.</p> <p>Default is 1.</p>
Pacing Size	No	<p>The pacing size. For information about setting the pacing size, see "Configuring Pacing" on page 110.</p> <p>Default is 0.</p>
Interpret as Rows	No	Controls whether the pacing size is expressed in kilobytes or number of rows.

Connection Attribute	Required	Description
Offload Processing	No	<p>Controls whether to use offload processing. Offload processes transfers PowerExchange bulk data processing from the source system to the PowerCenter Integration Service machine.</p> <p>Valid values:</p> <ul style="list-style-type: none"> - No. Disables offload processing. - Auto. PowerExchange determines whether to use offload processing. - Filter Before. Offloads processing to the PowerCenter Integration Service machine but continues to filter data on the source system. <p>If you select Filter Before for an IMS unload file data source or a DB2 for z/OS image copy data source, PowerExchange changes the selection to Filter After.</p> <ul style="list-style-type: none"> - Filter After. Offloads the bulk data processing to the PowerCenter Integration Service machine, including the filtering of data. <p>Default is No.</p>
Worker Threads	No	<p>The number of threads that PowerExchange uses on the PowerCenter Integration Service machine to process bulk data.</p> <p>For optimal performance, this value should not exceed the number of installed or available processors on the Integration Service machine.</p> <p>Valid values are from 1 through 64.</p> <p>Default is 0, which disables multithreading. If you use reader or writer pipeline partitioning, accept the default value of 0. You cannot use both multiple worker threads and partitioning.</p> <p>Not all PWX NRDB Batch connection types support worker threads. If the Worker Threads connection attribute for one of these connections is set to a nonzero value, processing continues without threads.</p> <p>Message PWX-31505 reports connection performance settings, including the number or worker threads. If worker threads are not used, message PWX-31505 reports a zero value for worker threads.</p>
Array Size	No	<p>For VSAM data sets and sequential files, the size of the storage array, in number of records, that is used for partitioned or multithreaded sessions.</p> <p>For partitioned sessions, this array size is shared across the partitions. For multithreaded sessions, each thread has this array size.</p> <p>For DB2 for z/OS image copies, determines the maximum memory buffer that might need to be allocated. If zIIP processing is enabled, Array Size also specifies how many compressed rows to dispatch to the zIIP processor.</p> <p>Valid values are from 1 through 5000. Default is 25.</p> <p>To tune partitioned sessions, particularly when the Write Mode attribute specifies Confirm Write On, increase the array size.</p>
Write Mode	No	<p>Controls whether data is sent to the PowerExchange Listener synchronously or asynchronously.</p> <p>For information about Write Mode options, see "Configuring Write Mode" on page 112.</p> <p>Default is Confirm Write On.</p>
Convert character data to string	No	<p>Controls whether character fields are converted to string fields so that embedded nulls in data are processed as spaces.</p> <p>By default, this attribute is not selected.</p> <p>For more information, see "Converting Character Data to Strings" on page 110.</p>

Connection Attribute	Required	Description
Retrieve PWX Log Entries	No	<p>Controls whether PWXPC writes PowerExchange informational messages related to the session in the PowerCenter session log.</p> <p>Default is to write PowerExchange error and warning messages but not informational messages to the session log.</p>

Connection Attribute	Required	Description
PWX Override	No	<p>PowerExchange connection overrides, separated by a semicolon. You can enter one or more of the following optional overrides:</p> <ul style="list-style-type: none"> - APPBUFSIZE=<i>app_buf_size</i>. Specifies the initial size of the application buffer for the connection. Overrides the APPBUFSIZE statement in the DBMOVER configuration file. - CNV_ERROR_OK= {N Y}. If you use a NRDB batch connection to write data to a nonrelational target, set this parameter to Y to allow the workflow continue after a PWX-00143 message reports a DBAPI illegal null conversion error. This error occurs when the workflow tries to write null data to a target NOT NULL column. Set this parameter to N to have the workflow end abnormally when this type of error occurs. Default is N. - CONNECTSTRINGCODEPAGE=<i>code_page</i>. Code page of the characters in the connection string. Enter this override if PowerExchange issues message PWX-07610. - CSQ_MEMSIZE=<i>memory_size</i>. The memory size, in megabytes, for the cache that is used for multiple-record writes before it spills to disk. Default is 16. - LOWVALUES={Y N}. Indicates whether PowerExchange preserves hexadecimal '0' values, called <i>low values</i>, in source character fields when passing these values to a PowerCenter session. Enter Y to preserve low values. When the session runs, PWXPC can write these values to a VSAM target on z/OS or to a sequential file target on Linux, UNIX, Windows, or z/OS. Overrides the LOWVALUES statement in the DBMOVER configuration file. - PWXNOQUOTES={Y N}. By default, in the SQL SELECT statements that PWXPC uses to select data for session processing, PWXPC adds double-quotation marks (") around every table name and field name that includes spaces or special characters. These special characters are /+~=-~`!%&*() [] {} ' ; ? , < > \ . If a table is identified by both a schema name and table name in the format <i>schema.table_name</i>, PWXPC places the quotation marks only around the "table_name" value. If for any reason, you cannot use quoted table names or field names in your environment, set PWXNOQUOTES to Y to override the default behavior. - RETLOGINFOMSG={N Y}. By default, PWXPC writes PowerExchange error and warning messages but not informational messages to the session log. If you specify RETLOGINFOMSG=Y, PWXPC writes informational messages as well as error and warning messages to the session log. Note: The RETLOGINFOMSG connection override is equivalent to the Retrieve PWX Log Entries connection attribute. - TCP/IP_OP_TIMEOUT=<i>network_operation_timeout</i>. A network operation timeout interval in seconds. For more information about this timeout override, see "Asynchronous Network Communication" on page 127. - TCP/IP_CON_TIMEOUT=<i>connection_timeout</i>. A connection timeout interval in seconds. For more information about this timeout override, see "Asynchronous Network Communication" on page 127. - WRT_ERROR_HANDLING=Y. If you use writer partitioning and set the Write Mode attribute to Confirm Write On, specify WRT_ERROR_HANDLING=Y to use an alternative method of error handling that is more efficient than the default method when the input data contains many errors. You can also specify this statement in the DBMOVER configuration file.
Connection Retry Period	No	<p>The number of seconds the PowerCenter Integration Service attempts to reconnect to the PowerExchange Listener after the initial connection attempt fails. If the Integration Service cannot connect to the PowerExchange Listener within the retry period, the session fails.</p>

Connection Attribute	Required	Description
		Default value is 0, which disables connection retries based on this PWXPC connection attribute Note: PowerExchange Listener connections to source databases do not have connection resilience.

RELATED TOPICS:

- [“Configuring Pacing” on page 110](#)
- [“Converting Character Data to Strings” on page 110](#)
- [“Configuring Write Mode” on page 112](#)
- [“Retrieving PowerExchange Log Entries” on page 111](#)

NRDB CDC Application Connections

Configure NRDB CDC application connections to extract change data from nonrelational sources in real-time extraction mode or from PowerExchange Condense condense files in batch extraction mode.

The following table describes the connection attributes for the PWX NRDB CDC Change and CDC Real Time application connection types:

Connection Attribute	Required?	Change or Real Time	Description
Name	Yes	Both	A name for the application connection.
Code Page	Yes	Both	The code page that the PowerCenter Integration Service uses to extract data from the source database. Note: In Unicode mode, PWXPC sets the code page with this value overriding any code page specification in the PowerExchange configuration file.
Location	Yes	Both	The location of the source or target database, as specified in a NODE statement in the DBMOVER configuration file.
User Name	Yes	Both	A user name for connecting to the database. For connections to supported Linux, UNIX, or Windows platforms, if you have enabled PowerExchange LDAP user authentication, the user name is the enterprise user name. For more information, see the <i>PowerExchange Reference Manual</i> .

Connection Attribute	Required?	Change or Real Time	Description
Password	Yes	Both	<p>A password for the specified user or a valid PowerExchange passphrase. Passwords that are less than nine characters in length are not validated. To avoid errors that might arise due to code page differences, Informatica recommends that the password includes only those characters that are allowed for passphrases.</p> <p>A PowerExchange passphrase for z/OS access can be from 9 to 128 characters in length when you use a PWXPC connection. A passphrase can contain the following characters:</p> <ul style="list-style-type: none"> - Uppercase and lowercase letters - The numbers 0 to 9 - Spaces - The following special characters: ' - ; # \ , . / ! % & * () _ + { } : @ < > ? <p>Note: The first character is an apostrophe.</p> <p>Passphrases cannot include single quotation marks ('), double quotation marks ("), or currency symbols.</p> <p>The allowable characters in the IBM IRRPHREX exit do not affect the allowable characters in PowerExchange passphrases.</p> <p>Note: 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.</p> <p>To use passphrases, ensure that the PowerExchange Listener runs with a security setting of SECURITY=(1,N) or higher in the DBMOVER member. For more information, see "SECURITY Statement" in the <i>PowerExchange Reference Manual</i>.</p> <p>To use passphrases for IMS connections, ensure that the following additional requirements are met:</p> <ul style="list-style-type: none"> - You must configure ODBA access to IMS as described in the <i>PowerExchange Navigator User Guide</i>. - 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.
Compression	No	Both	Controls whether to compress source data during the PowerCenter session.
Encryption Type	Yes	Both	<p>The encryption type. For information about supported encryption types, see "Configuring Encryption and Compression" on page 109.</p> <p>Default is None.</p>
Encryption Level	No	Both	<p>The encryption level. For information about supported encryption levels, see "Configuring Encryption and Compression" on page 109.</p> <p>Default is 1.</p>
Pacing Size	Yes	Both	The pacing size.
Interpret as Rows	No	Both	Indicates whether the pacing size is in number of rows.

Connection Attribute	Required?	Change or Real Time	Description
Image Type	No	Both	Indicates how PWXPC passes captured Updates to CDC sessions: as Updates, or as Deletes followed by Inserts. Default is BA, which processes Updates as Deletes followed by Inserts. For more information, see "Configuring Image Type" on page 113 .
Application Name	No	Both	The application name.
RestartToken File Folder	Yes	Both	The name of folder that contains the restart token folder. Default is \$PMRootDir/Restart.
RestartToken File Name	No	Both	The name of the restart token file.
Number of Runs to Keep RestartToken File	No	Both	The maximum number of backup copies to keep of the restart token file. Default is 0.
Recovery Cache Folder	No	Both	The name of the file cache folder that is used for recovery of the session. Default is \$PMRootDir/Cache.
UOW Count	No	Both	The number of units of work (UOWs) that PWXPC processes before it sends a commit request to PowerCenter to commit data to all targets in the CDC session. Valid values: -1. Disables this attribute. 0. Disables this attribute. <i>n</i> . Specifies the number of UOWs that PWXPC processes before sending a commit request. Default is 1. For more information, see "Configuring UOW Count" on page 118 .
Reader Time Limit	No	Real Time	The number of seconds that PowerCenter Integration Service reads data from the source before stopping. Default is 0, which indicates that the reader time is not limited. This value is intended for testing purposes only. Tip: Use Idle Time instead.

Connection Attribute	Required?	Change or Real Time	Description
Idle Time	No	Real Time	<p>The number of seconds the PowerExchange Listener remains idle after reaching the end of the change log, as indicated by message PWX-09967, before returning an end-of-file (EOF).</p> <p>Valid values:</p> <ul style="list-style-type: none"> -1. The EOF is never returned. The session runs continuously. 0. The EOF is returned at the end of log. Then the session terminates successfully. n. The number of seconds that the PowerExchange Listener remains idle before returning an EOF. <p>Default is -1.</p> <p>For more information, see "Configuring Idle Time" on page 116.</p>
Real-Time Flush Latency in milli-seconds	No	Real Time	<p>The number of milliseconds that must elapse before PWXPC sends a commit request to PowerCenter. When this period elapses, PWXPC continues to read the changes in the current UOW until it reaches the end of the UOW. Then PWXPC sends a commit request to PowerCenter so that the data can be committed to the targets.</p> <p>Valid values:</p> <ul style="list-style-type: none"> -1. Disables this attribute. 0 to 86400. Specifies the number of milliseconds that must elapse before PWXPC sends a commit request to PowerCenter. If you specify a value from 0 through 2000, PWXPC uses 2000 milliseconds (2 seconds). <p>Default is 0.</p> <p>For more information, see "Configuring PWX Latency and Real-Time Flush Latency" on page 119.</p>
PWX Latency in seconds	No	Real Time	<p>The maximum time, in seconds, that PowerExchange on the source platform waits for more change data before it flushes to PWXPC on the PowerCenter Integration Service platform.</p> <p>Default is 2.</p>
Maximum Rows Per commit	No	Both	<p>The maximum number of change records that PWXPC processes before it sends a commit request to PowerCenter to commit data to all targets in the CDC session. PWXPC does not wait for a UOW boundary to commit the change data.</p> <p>Default is 0, which disables this attribute.</p> <p>For more information, see "Configuring Maximum Rows per Commit" on page 120.</p>
Minimum Rows Per commit	No	Real Time	<p>The minimum number of change records that the PowerExchange Listener must read from the change stream before it passes a commit record to PWXPC. Before reaching this minimum, the PowerExchange Listener passes only change records to PWXPC and discards any commit records.</p> <p>Default is 0, which disables this attribute.</p> <p>For more information, see "Configuring Minimum Rows per Commit" on page 121.</p>
Convert character data to string	No	Both	<p>Controls whether to convert character fields to string fields so that embedded nulls in data are processed as spaces.</p> <p>By default, this attribute is not selected.</p>

Connection Attribute	Required?	Change or Real Time	Description
Event Table	No	Real Time	The PowerExchange extraction map name that is used for event processing. For more information, see “Configuring Event Table Processing” on page 114 .
Offload Processing	No	Real Time	Controls whether to use CDC offload processing to move PowerExchange processing for the change data from the source system to the PowerCenter Integration Service machine. Valid values: <ul style="list-style-type: none"> - No - Yes - Auto. PowerExchange determines whether to use offload processing. Default is No.
Worker Threads	No	Real Time	If you enable CDC offload processing, the number of threads that PowerExchange uses on the PowerCenter Integration Service machine to process change data. For optimal performance, the value should not exceed the number of installed or available processors on the PowerCenter Integration Service machine. Valid values are from 1 through 64. Default is 0.
Array Size	No	Real Time	If the Worker Threads value is greater than 0, the size of the storage array, in number of records, for the threads. Valid values are from 1 through 5000. Default is 25.
Map Location	No	Real Time	If you enable offload processing, the location of the source extraction maps. This value must be the name of a location in a NODE statement in the DBMOVER configuration file on the PowerCenter Integration Service machine.
Map Location User	No	Real Time	If you specified Map Location , a valid user ID for the user of that location.
Map Location Password	No	Real Time	If you specify Map Location User , a valid password for that user ID.
CAPI Connection Name Override	No	Real Time	A name that overrides the default CAPI connection name.
Retrieve PWX Log Entries	No	Both	Controls whether PWXPC writes PowerExchange informational messages related to a CDC session in the PowerCenter session log. Default is to write PowerExchange error and warning messages but not informational messages to the session log.

Connection Attribute	Required?	Change or Real Time	Description
PWX Override	No	Both	<p>PowerExchange connection override values, separated by a semicolon. You can enter one or more of the following overrides:</p> <ul style="list-style-type: none"> - APPBUFSIZE=<i>app_buf_size</i>. Specifies the initial size of the application buffer for the connection. Overrides the APPBUFSIZE statement in the DBMOVER configuration file. - CONNECTSTRINGCODEPAGE=<i>code_page</i>. Code page of the characters in the connection string. Enter this override if PowerExchange issues message PWX-07610. - LOWVALUES={Y N}. Indicates whether PowerExchange preserves hexadecimal '0' values, called <i>low values</i>, in source character fields when passing these values to a PowerCenter session. Enter Y to preserve low values. When the session runs, PWXPC can write these values to a VSAM target on z/OS or to a sequential file target on Linux, UNIX, Windows, or z/OS. Overrides the LOWVALUES statement in the DBMOVER configuration file. - PWXNOQUOTES={Y N}. By default, in the SQL SELECT statements that PWXPC uses to select data for session processing, PWXPC adds double-quotation marks (") around every table name and field name that includes spaces or special characters. These special characters are /+~=-~!%&*()[]{}';?,<>\\ . If a table is identified by both a schema name and table name in the format <i>schema.table_name</i>, PWXPC places the quotation marks only around the "<i>table_name</i>" value. If for any reason, you cannot use quoted table names or field names in your environment, set PWXNOQUOTES to Y to override the default behavior. - RETLOGINFOMSG={N Y}. By default, PWXPC writes PowerExchange error and warning messages but not informational messages to the session log. If you specify RETLOGINFOMSG=Y, PWXPC writes informational messages as well as error and warning messages to the session log. Note: The RETLOGINFOMSG connection override is equivalent to the Retrieve PWX Log Entries connection attribute. - TCPIP_OP_TIMEOUT=<i>network_operation_timeout</i>. A network operation timeout interval in seconds. For more information about this timeout override, see "Asynchronous Network Communication" on page 127. - TCPIP_CON_TIMEOUT=<i>connection_timeout</i>. A connection timeout interval in seconds. For more information about this timeout override, see "Asynchronous Network Communication" on page 127.
Connection Retry Period	No	Both	<p>The number of seconds the PowerCenter Integration Service attempts to reconnect to the PowerExchange Listener after the initial connection attempt fails. If the Integration Service cannot connect to the PowerExchange Listener within the retry period, the session fails.</p> <p>Default value is 0, which disables connection retries based on this PWXPC connection attribute.</p> <p>Note: PowerExchange Listener connections to source databases do not have connection resilience.</p>

For more information about CDC offload processing, see the *PowerExchange CDC Guide for z/OS*.

RELATED TOPICS:

- [“DB2 CDC Application Connections” on page 70](#)
- [“DB2 Relational Connections” on page 62](#)
- [“Microsoft SQL Server CDC Application Connections” on page 94](#)
- [“Microsoft SQL Server Relational Connections” on page 91](#)
- [“NRDB Batch Application Connections” on page 77](#)
- [“NRDB Lookup Relational Connections” on page 88](#)
- [“Oracle CDC Application Connections” on page 102](#)
- [“Oracle Relational Connections” on page 99](#)
- [“Configuring PowerExchange Connections” on page 61](#)
- [“Configuring Encryption and Compression” on page 109](#)
- [“Configuring Pacing” on page 110](#)
- [“Converting Character Data to Strings” on page 110](#)
- [“Retrieving PowerExchange Log Entries” on page 111](#)
- [“Configuring Image Type” on page 113](#)
- [“Configuring CDC Restart Attributes” on page 117](#)
- [“Configuring UOW Count” on page 118](#)
- [“Configuring Idle Time” on page 116](#)
- [“Configuring the CAPI Connection Name Override” on page 115](#)
- [“Configuring PWX Latency and Real-Time Flush Latency” on page 119](#)
- [“Configuring Maximum Rows per Commit” on page 120](#)
- [“Commit Processing for CDC Sessions” on page 157](#)
- [“Managing Session Log and Restart Token File History” on page 181](#)
- [“Configuring Event Table Processing” on page 114](#)
- [“Enabling Recovery for a CDC Session” on page 173](#)

NRDB Lookup Relational Connections

Configure NRDB Lookup relational connections if you use lookups to extract data from PowerExchange nonrelational sources.

The following table describes the connection attributes for the PWX NRDB Lookup relational connection type:

Connection Attribute	Required	Description
Name	Yes	Name of the relational connection.
Code Page	Yes	Select the code page for the PowerCenter Integration Service to use to extract data from the source database. Note: In Unicode mode, PWXPC sets the code page with this value overriding any code page specification in the PowerExchange configuration file.

Connection Attribute	Required	Description
Location	Yes	Location of the source or target database as specified as a node in the PowerExchange configuration file dbmover.cfg.
User Name	Yes	User name for the PowerExchange Listener to which you are connecting. For connections to a PowerExchange Listener on a supported Linux, UNIX, or Windows system, if you have enabled PowerExchange LDAP user authentication, the user name is the enterprise user name. For more information, see the <i>PowerExchange Reference Manual</i> .
Password	Yes	<p>A password for the specified user or a valid PowerExchange passphrase. Passwords that are less than nine characters in length are not validated.</p> <p>To avoid errors that might arise due to code page differences, Informatica recommends that the password includes only those characters that are allowed for passphrases.</p> <p>A PowerExchange passphrase for z/OS access can be from 9 to 128 characters in length when you use a PWXPC connection. A passphrase can contain the following characters:</p> <ul style="list-style-type: none"> - Uppercase and lowercase letters - The numbers 0 to 9 - Spaces - The following special characters: ' - ; # \ , . / ! % & * () _ + { } : @ < > ? <p>Note: The first character is an apostrophe.</p> <p>Passphrases cannot include single quotation marks ('), double quotation marks ("), or currency symbols.</p> <p>The allowable characters in the IBM IRRPHREX exit do not affect the allowable characters in PowerExchange passphrases.</p> <p>Note: 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.</p> <p>To use passphrases, ensure that the PowerExchange Listener runs with a security setting of SECURITY=(1,N) or higher in the DBMOVER member. For more information, see "SECURITY Statement" in the <i>PowerExchange Reference Manual</i>.</p> <p>To use passphrases for lookup connections to IMS, ensure that the following additional requirements are met:</p> <ul style="list-style-type: none"> - You must configure ODBA access to IMS as described in the <i>PowerExchange Navigator User Guide</i>. - 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.
Environment SQL	No	SQL commands run in the database environment.
Compression	No	Select to compress source data during the PowerCenter session.
Encryption Type	No	<p>The encryption type. For information about supported encryption types, see "Configuring Encryption and Compression" on page 109.</p> <p>Default is None.</p>

Connection Attribute	Required	Description
Encryption Level	No	The encryption level. For information about supported encryption levels, see “Configuring Encryption and Compression” on page 109 . Default is 1.
Pacing Size	No	Enter the pacing size.
Interpret as Rows	No	Specifies whether, or not, pacing size is in number of rows.
Convert character data to string	No	Convert character fields to string fields so that embedded nulls in data are processed as spaces. By default, this attribute is not selected.
PWX Override	No	Specifies PowerExchange connection override values, separated by semicolons, including the following overrides: <ul style="list-style-type: none"> - APPBUFSIZE=<i>app_buf_size</i>. Specifies the initial size of the application buffer for the connection. Overrides the APPBUFSIZE statement in the DBMOVER configuration file. - CONNECTSTRINGCODEPAGE=<i>code_page</i>. Code page of the characters in the connection string. Enter this override if PowerExchange issues message PWX-07610. - LOWVALUES={Y N}. Indicates whether PowerExchange preserves hexadecimal '0' values, called <i>low values</i>, in source character fields when passing these values to a PowerCenter session. Enter Y to preserve low values. When the session runs, PWXPC can write these values to a VSAM target on z/OS or to a sequential file target on Linux, UNIX, Windows, or z/OS. Overrides the LOWVALUES statement in the DBMOVER configuration file. - RETLOGINFOMSG={N Y}. By default, PWXPC writes PowerExchange error and warning messages but not informational messages to the session log. If you specify RETLOGINFOMSG=Y, PWXPC writes informational messages as well as error and warning messages to the session log. - TCPIP_OP_TIMEOUT=<i>network_operation_timeout</i>. A network operation timeout interval in seconds. For more information about this timeout override, see “Asynchronous Network Communication” on page 127. - TCPIP_CON_TIMEOUT=<i>connection_timeout</i>. A connection timeout interval in seconds. For more information about this timeout override, see “Asynchronous Network Communication” on page 127.
Connection Retry Period	No	The number of seconds the PowerCenter Integration Service attempts to reconnect to the PowerExchange Listener after the initial connection attempt fails. If the Integration Service cannot connect to the PowerExchange Listener within the retry period, the session fails. Default value is 0, which disables connection retries based on this PWXPC connection attribute. Note: PowerExchange Listener connections to source databases do not have connection resilience.

RELATED TOPICS:

- [“Configuring Encryption and Compression” on page 109](#)
- [“Configuring Pacing” on page 110](#)
- [“Converting Character Data to Strings” on page 110](#)

Microsoft SQL Server Relational Connections

Configure relational connections to extract data from and load data to Microsoft SQL Server sources and targets in bulk data movement sessions. Also use relational connections to apply change data to Microsoft SQL Server relational targets in CDC sessions.

The following table describes the connection attributes for the PWX MSSQLServer relational connection type:

Connection Attribute	Required	Description
Name	Yes	The name of the relational connection.
Code Page	Yes	The code page that the PowerCenter Integration Service uses to extract data from the source database. Note: In Unicode mode, PWXPC uses this code page instead of any code page specification in the PowerExchange dbmover.cfg configuration file.
Location	Yes	The location of the source or target database as specified in a NODE statement in the PowerExchange dbmover.cfg configuration file.
Server Name	Yes	The name of the SQL Server instance.
Database Name	Yes	The name of the SQL Server database.
User Name	Yes	A user name that can be used to access the database.
Password	Yes	The password for the specified user name.
Environment SQL	No	SQL commands that run in the database environment.
Compression	No	Select this option to compress source data during the PowerCenter session.
Encryption Type	No	The encryption type. For information about supported encryption types, see "Configuring Encryption and Compression" on page 109 . Default is None.
Encryption Level	No	The encryption level. For information about supported encryption levels, see "Configuring Encryption and Compression" on page 109 . Default is 1.
Pacing Size	No	The pacing size. For information about setting the pacing size, see "Configuring Pacing" on page 110 . Default is 0.
Interpret as Rows	No	Controls whether the pacing size is in number of rows.
Convert character data to string	No	Controls whether character fields are converted to string fields so that embedded nulls in data are processed as spaces. By default, this attribute is not selected. For more information, see "Converting Character Data to Strings" on page 110 .

Connection Attribute	Required	Description
Write Mode	No	The write mode. For information about Write Mode options, see “Configuring Write Mode” on page 112 . Default is Confirm Write On .
Array Size	No	The size of the storage array, in number of records, for SQL Server bulk load operations. Valid values are from 1 through 5000. Default is 25.
Reject File	No	Overrides the default prefix of PWXR for the reject file. PowerExchange creates the reject file on the target machine when Write Mode is set to Asynchronous with Fault Tolerance . Note: Enter PWXDISABLE to prevent creation of the reject files.

Connection Attribute	Required	Description
PWX Override	No	<p>PowerExchange connection overrides, separated by a semicolon. You can enter one or more of the following optional overrides:</p> <ul style="list-style-type: none"> - APPBUFSIZE=<i>app_buf_size</i>. Specifies the initial size of the application buffer for the connection. Overrides the APPBUFSIZE statement in the DBMOVER configuration file. - CONNECTSTRINGCODEPAGE=<i>code_page</i>. Code page of the characters in the connection string. Enter this override if PowerExchange issues message PWX-07610. - LOWVALUES={Y N}. Indicates whether PowerExchange preserves hexadecimal '0' values, called <i>low values</i>, in source character fields when passing these values to a PowerCenter session. Enter Y to preserve low values. When the session runs, PWXPC can write these values to a VSAM target on z/OS or to a sequential file target on Linux, UNIX, Windows, or z/OS. Overrides the LOWVALUES statement in the DBMOVER configuration file. - PWXNOQUOTES={Y N}. By default, in the SQL SELECT statements that PWXPC uses to select data for session processing, PWXPC adds double-quotation marks (") around every table name and field name that includes spaces or special characters. These special characters are /+--~`!%^&*()[]{}';?,<>\\ . If a table is identified by both a schema name and table name in the format <i>schema.table_name</i>, PWXPC places the quotation marks only around the <i>table_name</i> value. If for any reason, you cannot use quoted table names or field names in your environment, set PWXNOQUOTES to Y to override the default behavior. - RETLOGINFOMSG={N Y}. By default, PWXPC writes PowerExchange error and warning messages but not informational messages to the session log. If you specify RETLOGINFOMSG=Y, PWXPC writes informational messages as well as error and warning messages to the session log. - TCPIP_OP_TIMEOUT=<i>network_operation_timeout</i>. A network operation timeout interval in seconds. For more information about this timeout override, see "Asynchronous Network Communication" on page 127. - TCPIP_CON_TIMEOUT=<i>connection_timeout</i>. A connection timeout interval in seconds. For more information about this timeout override, see "Asynchronous Network Communication" on page 127.
Connection Retry Period	No	<p>The number of seconds the PowerCenter Integration Service attempts to reconnect to the PowerExchange Listener after the initial connection attempt fails. If the Integration Service cannot connect to the PowerExchange Listener within the retry period, the session fails.</p> <p>Default value is 0, which disables connection retries based on this PWXPC connection attribute.</p> <p>Note: PowerExchange Listener connections to source databases do not have connection resilience.</p>

RELATED TOPICS:

- ["Configuring Encryption and Compression" on page 109](#)
- ["Configuring Pacing" on page 110](#)
- ["Converting Character Data to Strings" on page 110](#)
- ["Configuring Write Mode" on page 112](#)

Microsoft SQL Server CDC Application Connections

Configure CDC application connections to extract change data from Microsoft SQL Server sources in real-time extraction mode or from PowerExchange Logger for Linux, UNIX, and Windows log files in batch or continuous extraction mode.

The following table describes the connection attributes for the MSSQL CDC Change and CDC Real Time application connection types:

Connection Attribute	Required?	Change or Real Time	Description
Name	Yes	Both	The name of the application connection.
Code Page	Yes	Both	The code page that the PowerCenter Integration Service uses to extract data from the source database. Note: In Unicode mode, PWXPC uses this code page instead of any code page specification in the PowerExchange dbmover.cfg configuration file.
Location	Yes	Both	The location of the source or target database as specified in a NODE statement in the PowerExchange dbmover.cfg configuration file.
User Name	Yes	Both	A user name that can be used to access the SQL Server database.
Password	Yes	Both	The password for the specified user name.
Compression	No	Both	Select this option to compress source data during the PowerCenter session.
Encryption Type	Yes	Both	The encryption type. For information about encryption types, see "Configuring Encryption and Compression" on page 109 . Default is None.
Encryption Level	No	Both	The encryption level. For information about encryption levels, see "Configuring Encryption and Compression" on page 109 . Default is 1.
Pacing Size	Yes	Both	The pacing size. For information about setting the pacing size, see "Configuring Pacing" on page 110 . Default is 0.
Interpret as Rows	No	Both	Controls whether the pacing size is in number of rows.
Image Type	No	Both	Indicates whether PWXPC passes captured Update operations to CDC sessions as Updates, or as Deletes followed by Inserts. Default is BA, which processes Updates as Deletes followed by Inserts. For more information, see "Configuring Image Type" on page 113 .
Application Name	No	Both	The application name.

Connection Attribute	Required?	Change or Real Time	Description
RestartToken File Folder	Yes	Both	The restart token folder. Default is \$PMRootDir/Restart. For more information, see “Configuring CDC Restart Attributes” on page 117 .
RestartToken File Name	No	Both	The restart token file name.
Number of Runs to Keep RestartToken File	No	Both	The maximum number of backup copies of the restart token file to keep. Default is 0.
Recovery Cache Folder	No	Both	The file cache folder to enable recovery for the session. Default is \$PMRootDir/Cache.
UOW Count	No	Both	The number of units of work (UOWs) that PWXPC processes before it sends a commit request to PowerCenter to commit data to all targets in the CDC session. Valid values: -1. Disables this attribute. 0. Disables this attribute. <i>n</i> . Specifies the number of UOWs that PWXPC processes before sending a commit request. Default is 1. For more information, see “Configuring UOW Count” on page 118 .
Reader Time Limit	No	Real Time	The number of seconds that the PowerCenter Integration Service reads data from the source before stopping. If you enter 0, the Reader Time Limit attribute does not limit the reader time. This attribute is intended for testing purposes only. Tip: You can use Idle Time instead of Reader Time Limit . Default is 0.
Idle Time	No	Real Time	The number of seconds that the PowerExchange Listener remains idle after reaching the end of the change log, as indicated by message PWX-09967, before returning an end-of-file (EOF). Valid values: -1. The EOF is not returned. The session runs continuously. 0. The EOF is returned at the end of log. The session terminates successfully. <i>n</i> . The number of seconds after which the EOF is returned. Default is -1. For more information, see “Configuring Idle Time” on page 116 .

Connection Attribute	Required?	Change or Real Time	Description
Real-Time Flush Latency in milliseconds	No	Real Time	<p>The number of milliseconds that must elapse before PWXPC sends a commit request to PowerCenter. When this period elapses, PWXPC continues to read the changes in the current UOW until it reaches the end of the UOW. Then PWXPC sends a commit request to PowerCenter so that the data can be committed to the targets.</p> <p>Valid values:</p> <p>-1. Disables this attribute.</p> <p>0 to 86400. Specifies the number of milliseconds that must elapse before PWXPC sends a commit request to PowerCenter. If you specify a value from 0 through 2000, PWXPC uses 2000 milliseconds (2 seconds). Default is 0.</p> <p>For more information, see “Configuring PWX Latency and Real-Time Flush Latency” on page 119.</p>
PWX Latency in seconds	No	Real Time	<p>The maximum number of seconds that PowerExchange extraction processing on the source platform waits for more change data before it returns control to PWXPC on the PowerCenter Integration Service machine.</p> <p>Default is 2.</p>
Maximum Rows Per commit	No	Both	<p>The maximum number of change records that PWXPC processes before it sends a commit request to PowerCenter to commit data to all targets in the CDC session. PWXPC does not wait for a UOW boundary to commit the change data.</p> <p>Default is 0, which disables this attribute.</p> <p>For more information, see “Configuring Maximum Rows per Commit” on page 120.</p>
Minimum Rows Per commit	No	Real Time	<p>The minimum number of change records that the PowerExchange Listener must read from the change stream before it passes a commit record to PWXPC. Before reaching this minimum, the PowerExchange Listener passes only change records to PWXPC and discards any commit records.</p> <p>Default is 0, which disables this attribute.</p> <p>For more information, see “Configuring Minimum Rows per Commit” on page 121.</p>
Convert character data to string	No	Both	<p>Controls whether to convert character fields to string fields so that embedded nulls in data are processed as spaces.</p> <p>By default, this attribute is not selected.</p> <p>For more information, see “Converting Character Data to Strings” on page 110.</p>
Event Table	No	Real Time	<p>The PowerExchange extraction map name that is used for event processing.</p> <p>For more information, see “Configuring Event Table Processing” on page 114.</p>

Connection Attribute	Required?	Change or Real Time	Description
Offload Processing	No	Real Time	<p>Controls whether PowerExchange uses CDC offload processing. When offload processing is enabled, PowerExchange transfers column-level processing of the change data from the source system to the PowerCenter Integration Service machine.</p> <p>Valid values:</p> <ul style="list-style-type: none"> - No. Disables offload processing. - Yes. Enables offload processing. - Auto. PowerExchange determines whether to use offload processing. <p>Default is No.</p> <p>For more information about offload processing, see the <i>PowerExchange CDC Guide for Linux, UNIX, and Windows</i>.</p>
Worker Threads	No	Real Time	<p>If you use CDC offload processing, the number of threads that PowerExchange uses on the PowerCenter Integration Service machine to process change data.</p> <p>For optimal performance, this value should not exceed the number of installed or available processors on the Integration Service machine.</p> <p>Valid values are from 1 through 64.</p> <p>Default is 0.</p>
Array Size	No	Real Time	<p>If the Worker Threads value is greater than 0, specifies the size of the storage array, in number of records, for the threads.</p> <p>Valid values are from 1 through 5000.</p> <p>Default is 25.</p>
Map Location	No	Real Time	<p>If you use remote logging to a PowerExchange Logger for Linux, UNIX, and Windows, the location of the source extraction maps. This value must be the node name of the location, as specified in a NODE statement in the PowerExchange dbmover.cfg configuration file on the Integration Service machine.</p> <p>For more information about remote logging, see the <i>PowerExchange CDC Guide for Linux, UNIX, and Windows</i>.</p>
Map Location User	No	Real Time	<p>If you specify a Map Location value, a valid user ID for the map location.</p>
Map Location Password	No	Real Time	<p>If you specify a Map Location value, a valid password for the specified map location user.</p>
CAPI Connection Name Override	No	Real Time	<p>Overrides the default CAPI connection name.</p> <p>For more information, see “Configuring the CAPI Connection Name Override” on page 115.</p>
Logger DBID	No	Real Time	<p>The DBID parameter value that is specified in the PowerExchange Logger for Linux, UNIX, and Windows configuration file, pwxocl.</p> <p>This value is required only if the PowerExchange Logger extracts change data for articles in multiple publication databases. In this case, you must also set the MULTIPUB parameter to Y in the MSQ CAPI_CONNECTION statement in the PowerExchange dbmover.cfg configuration file. Otherwise, the extraction fails.</p>

Connection Attribute	Required?	Change or Real Time	Description
Retrieve PWX Log Entries	No	Both	Controls whether PWXPC writes PowerExchange informational messages related to a CDC session in the PowerCenter session log. Default is to write PowerExchange error and warning messages but not informational messages to the session log.
PWX Override	No	Both	<p>PowerExchange connection overrides, separated by a semicolon. You can enter one or more of the following optional overrides:</p> <ul style="list-style-type: none"> - APPBUFSIZE=<i>app_buf_size</i>. Specifies the initial size of the application buffer for the connection. Overrides the value of APPBUFSIZE in the DBMOVER configuration file. - CONNECTSTRINGCODEPAGE=<i>code_page</i>. Code page of the characters in the connection string. Enter this override if PowerExchange issues message PWX-07610. - LOWVALUES={Y N}. Indicates whether PowerExchange preserves hexadecimal '0' values, called <i>low values</i>, in source character fields when passing these values to a PowerCenter session. Enter Y to preserve low values. When the session runs, PWXPC can write these values to a VSAM target on z/OS or to a sequential file target on Linux, UNIX, Windows, or z/OS. Overrides the LOWVALUES statement in the DBMOVER configuration file. - PWXNOQUOTES={Y N}. By default, in the SQL SELECT statements that PWXPC uses to select data for session processing, PWXPC adds double-quotation marks (") around every table name and field name that includes spaces or special characters. These special characters are /+--~`!%^&*()[]{}';?,<>\\ . If a table is identified by both a schema name and table name in the format <i>schema.table_name</i>, PWXPC places the quotation marks only around the "<i>table_name</i>" value. If for any reason, you cannot use quoted table names or field names in your environment, set PWXNOQUOTES to Y to override the default behavior. - RETLOGINFOMSG={N Y}. By default, PWXPC writes PowerExchange error and warning messages but not informational messages to the session log. If you specify RETLOGINFOMSG=Y, PWXPC writes informational messages as well as error and warning messages to the session log. Note: The RETLOGINFOMSG connection override is equivalent to the Retrieve PWX Log Entries connection attribute. - TCPIP_OP_TIMEOUT=<i>network_operation_timeout</i>. A network operation timeout interval in seconds. For more information about this timeout override, see "Asynchronous Network Communication" on page 127. - TCPIP_CON_TIMEOUT=<i>connection_timeout</i>. A connection timeout interval in seconds. For more information about this timeout override, see "Asynchronous Network Communication" on page 127.
Connection Retry Period	No	Both	<p>The number of seconds the PowerCenter Integration Service attempts to reconnect to the PowerExchange Listener after the initial connection attempt fails. If the Integration Service cannot connect to the PowerExchange Listener within the retry period, the session fails.</p> <p>Default value is 0, which disables connection retries based on this PWXPC connection attribute.</p> <p>Note: PowerExchange Listener connections to source databases do not have connection resilience.</p>

RELATED TOPICS:

- [“Configuring Encryption and Compression” on page 109](#)
- [“Configuring Pacing” on page 110](#)
- [“Converting Character Data to Strings” on page 110](#)
- [“Retrieving PowerExchange Log Entries” on page 111](#)
- [“Configuring Image Type” on page 113](#)
- [“Configuring Idle Time” on page 116](#)
- [“Configuring the CAPI Connection Name Override” on page 115](#)
- [“Configuring UOW Count” on page 118](#)
- [“Configuring CDC Restart Attributes” on page 117](#)
- [“Configuring PWX Latency and Real-Time Flush Latency” on page 119](#)
- [“Configuring Maximum Rows per Commit” on page 120](#)
- [“Commit Processing for CDC Sessions” on page 157](#)
- [“Managing Session Log and Restart Token File History” on page 181](#)
- [“Enabling Recovery for a CDC Session” on page 173](#)

Oracle Relational Connections

Configure relational connections to extract data from and load data to Oracle sources and targets in bulk data movement sessions. Also use relational connections to apply change data to Oracle relational targets in CDC sessions.

The following table describes the connection attributes for the PWX Oracle relational connection type:

Connection Attribute	Required	Description
Name	Yes	The name of the relational connection.
Code Page	Yes	The code page that the PowerCenter Integration Service uses to extract data from the source database. Note: In Unicode mode, PWXPC uses this code page instead of any code page specification in the PowerExchange dbmover.cfg configuration file.
Location	Yes	The location of the source or target database as specified in a NODE statement in the PowerExchange dbmover.cfg configuration file.
TNS Name	Yes	The Net Service Name of the Oracle instance.
User Name	Yes	A user name that can be used to access the database. For databases on supported Linux, UNIX, or Windows systems, if you have enabled PowerExchange LDAP user authentication and disabled relational pass-through authentication, the user name is the enterprise user name. For more information, see the <i>PowerExchange Reference Manual</i> .
Password	Yes	The password for the specified user name.

Connection Attribute	Required	Description
Environment SQL	No	SQL commands that run in the database environment.
Compression	No	Select this option to compress source data during the PowerCenter session.
Encryption Type	No	The encryption type. For information about supported encryption types, see “Configuring Encryption and Compression” on page 109 . Default is None.
Encryption Level	No	The encryption level. For information about supported encryption levels, see “Configuring Encryption and Compression” on page 109 .
Pacing Size	No	The pacing size. For information about setting the pacing size, see “Configuring Pacing” on page 110 . Default is 0.
Interpret as Rows	No	Controls whether the pacing size is in number of rows.
Convert character data to string	No	Controls whether character fields are converted to string fields so that embedded nulls in data are processed as spaces. By default, this attribute is not selected. For more information, see “Converting Character Data to Strings” on page 110 .
Write Mode	No	The write mode. For information about Write Mode options, see “Configuring Write Mode” on page 112 . Default is Confirm Write On .
Reject File	No	Overrides the default prefix of PWXR for the reject file. PowerExchange creates the reject file on the target machine when Write Mode is set to Asynchronous with Fault Tolerance . Note: Enter PWXDISABLE to prevent creation of the reject files.

Connection Attribute	Required	Description
PWX Override	No	<p>PowerExchange connection overrides, separated by a semicolon. You can enter one or more of the following optional overrides:</p> <ul style="list-style-type: none"> - APPBUFSIZE=<i>app_buf_size</i>. Specifies the initial size of the application buffer for the connection. Overrides the APPBUFSIZE statement in the DBMOVER configuration file. - CONNECTSTRINGCODEPAGE=<i>code_page</i>. Code page of the characters in the connection string. Enter this override if PowerExchange issues message PWX-07610. - LOWVALUES={Y N}. Indicates whether PowerExchange preserves hexadecimal '0' values, called <i>low values</i>, in source character fields when passing these values to a PowerCenter session. Enter Y to preserve low values. When the session runs, PWXPC can write these values to a VSAM target on z/OS or to a sequential file target on Linux, UNIX, Windows, or z/OS. Overrides the LOWVALUES statement in the DBMOVER configuration file. - PWXNOQUOTES={Y N}. By default, in the SQL SELECT statements that PWXPC uses to select data for session processing, PWXPC adds double-quotation marks (") around every table name and field name that includes spaces or special characters. These special characters are /+--~`!%&*()[]{}';?,<>\\ . If a table is identified by both a schema name and table name in the format <i>schema.table_name</i>, PWXPC places the quotation marks only around the <i>table_name</i> value. If for any reason, you cannot use quoted table names or field names in your environment, set PWXNOQUOTES to Y to override the default behavior. - RETLOGINFOMSG={N Y}. By default, PWXPC writes PowerExchange error and warning messages but not informational messages to the session log. If you specify RETLOGINFOMSG=Y, PWXPC writes informational messages as well as error and warning messages to the session log. - TCPIP_OP_TIMEOUT=<i>network_operation_timeout</i>. A network operation timeout interval in seconds. For more information about this timeout override, see "Asynchronous Network Communication" on page 127. - TCPIP_CON_TIMEOUT=<i>connection_timeout</i>. A connection timeout interval in seconds. For more information about this timeout override, see "Asynchronous Network Communication" on page 127.
Connection Retry Period	No	<p>The number of seconds the PowerCenter Integration Service attempts to reconnect to the PowerExchange Listener after the initial connection attempt fails. If the Integration Service cannot connect to the PowerExchange Listener within the retry period, the session fails.</p> <p>Default value is 0, which disables connection retries based on this PWXPC connection attribute.</p> <p>Note: PowerExchange Listener connections to source databases do not have connection resilience.</p>

For more information, see the *PowerExchange Bulk Data Movement Guide*.

RELATED TOPICS:

- ["Configuring Encryption and Compression" on page 109](#)
- ["Configuring Pacing" on page 110](#)
- ["Converting Character Data to Strings" on page 110](#)
- ["Configuring Write Mode" on page 112](#)

Oracle CDC Application Connections

Configure CDC application connections to extract change data from Oracle sources in real-time extraction mode or from PowerExchange Logger for Linux, UNIX, and Windows log files in batch or continuous extraction mode.

The following table describes the connection attributes for the PWX Oracle CDC application connection type:

Connection Attribute	Required?	Change or Real Time	Description
Name	Yes	Both	The name of the application connection.
Code Page	Yes	Both	The code page that the PowerCenter Integration Service uses to extract data from the source database. In Unicode mode, you must select UTF-8 for Oracle CDC connections. Note: In Unicode mode, PWXPC sets the code page with this value overriding any code page specification in the PowerExchange configuration file.
Location	Yes	Both	The location of the source or target database as specified in a NODE statement in the PowerExchange dbmover.cfg configuration file.
User Name	Yes	Both	A user name that can be used to access the Oracle database. For a database on a supported Linux, UNIX, or Windows system, if you have enabled PowerExchange LDAP user authentication, the user name is the enterprise user name. For more information, see the <i>PowerExchange Reference Manual</i> .
Password	Yes	Both	The password for the specified user name.
Compression	No	Both	Select this option to compress source data during the PowerCenter session.
Encryption Type	Yes	Both	The encryption type. For information about encryption types, see "Configuring Encryption and Compression" on page 109 . Default is None.
Encryption Level	No	Both	The encryption level. For information about encryption levels, see "Configuring Encryption and Compression" on page 109 . Default is 1.
Pacing Size	Yes	Both	The pacing size. For information about setting the pacing size, see "Configuring Pacing" on page 110 . Default is 0.
Interpret as Rows	No	Both	Controls whether the pacing size is in number of rows.
Image Type	No	Both	Indicates whether PWXPC passes captured Update operations to CDC sessions as Updates, or as Deletes followed by Inserts. Default is BA, which processes Updates as Deletes followed by Inserts. For more information, see "Configuring Image Type" on page 113 .

Connection Attribute	Required?	Change or Real Time	Description
Application Name	No	Both	The application name.
RestartToken File Folder	Yes	Both	The restart token folder. Default is \$PMRootDir/Restart. For more information, see "Configuring CDC Restart Attributes" on page 117 .
RestartToken File Name	No	Both	The restart token file name.
Number of Runs to Keep RestartToken File	No	Both	The maximum number of backup copies of the restart token file to keep. Default is 0.
Recovery Cache Folder	No	Both	The file cache folder to enable recovery for the session. Default is \$PMRootDir/Cache.
UOW Count	No	Both	The number of units of work (UOWs) that PWXPC processes before it sends a commit request to PowerCenter to commit data to all targets in the CDC session. Valid values: -1. Disables this attribute. 0. Disables this attribute. <i>n</i> . Specifies the number of UOWs that PWXPC processes before sending a commit request. Default is 1. For more information, see "Configuring UOW Count" on page 118 .
Reader Time Limit	No	Real Time	The number of seconds that the PowerCenter Integration Service reads data from the source before stopping. If you enter 0, the Reader Time Limit attribute does not limit the reader time. This attribute is intended for testing purposes only. Tip: Use Idle Time instead of Reader Time Limit . Default is 0.
Idle Time	No	Real Time	The number of seconds that the PowerExchange Listener remains idle after reaching the end of the change log, as indicated by message PWX-09967, before returning an end-of-file (EOF). Valid values: -1. The EOF is not returned. The session runs continuously. 0. The EOF is returned at the end of log. The session terminates successfully. <i>n</i> . The number of seconds after which the EOF is returned. Default is -1. For more information, see "Configuring Idle Time" on page 116 .

Connection Attribute	Required?	Change or Real Time	Description
Real-Time Flush Latency in milli-seconds	No	Real Time	<p>The number of milliseconds that must elapse before PWXPC sends a commit request to PowerCenter. When this period elapses, PWXPC continues to read the changes in the current UOW until it reaches the end of the UOW. Then PWXPC sends a commit request to PowerCenter so that the data can be committed to the targets.</p> <p>Valid values:</p> <p>-1. Disables this attribute.</p> <p>0 to 86400. Specifies the number of milliseconds that must elapse before PWXPC sends a commit request to PowerCenter. If you specify a value from 0 through 2000, PWXPC uses 2000 milliseconds (2 seconds). Default is 0.</p> <p>For more information, see "Configuring PWX Latency and Real-Time Flush Latency" on page 119.</p>
PWX Latency in seconds	No	Real Time	<p>The maximum number of seconds that PowerExchange extraction processing on the source platform waits for more change data before it returns control to PWXPC on the PowerCenter Integration Service machine.</p> <p>Default is 2.</p>
Maximum Rows Per commit	No	Both	<p>The maximum number of change records that PWXPC processes before it sends a commit request to PowerCenter to commit data to all targets in the CDC session. PWXPC does not wait for a UOW boundary to commit the change data.</p> <p>Default is 0, which disables this attribute.</p> <p>For more information, see "Configuring Maximum Rows per Commit" on page 120.</p>
Minimum Rows Per commit	No	Real Time	<p>The minimum number of change records that the PowerExchange Listener must read from the change stream before it passes a commit record to PWXPC. Before reaching this minimum, the PowerExchange Listener passes only change records to PWXPC and discards any commit records.</p> <p>Default is 0, which disables this attribute.</p> <p>For more information, see "Configuring Minimum Rows per Commit" on page 121.</p>
Instance Name	No	Real Time	<p>Overrides the Oracle instance name in the ORACLEID statement in the PowerExchange dbmover.cfg configuration file.</p>
Connect String	No	Real Time	<p>Overrides the connection value in the ORACLEID statement in the PowerExchange dbmover.cfg configuration file.</p>
Source Schema Name	No	Both	<p>Overrides the schema name for a group of registrations.</p>
Convert character data to string	No	Both	<p>Controls whether to convert character fields to string fields so that embedded nulls in data are processed as spaces.</p> <p>By default, this attribute is not selected.</p> <p>For more information, see "Converting Character Data to Strings" on page 110.</p>

Connection Attribute	Required?	Change or Real Time	Description
Event Table	No	Real Time	The PowerExchange extraction map name that is used for event processing. For more information, see "Configuring Event Table Processing" on page 114 .
Offload Processing	No	Real Time	Controls whether PowerExchange uses CDC offload processing. When offload processing is enabled, PowerExchange transfers column-level processing of the change data from the source system to the PowerCenter Integration Service machine. Valid values: <ul style="list-style-type: none"> - No. Disables offload processing. - Yes. Enables offload processing. - Auto. PowerExchange determines whether to use offload processing. Default is No. For more information about offload processing, see the <i>PowerExchange CDC Guide for Linux, UNIX, and Windows</i> .
Worker Threads	No	Real Time	If you use CDC offload processing, the number of threads that PowerExchange uses on the PowerCenter Integration Service machine to process change data. For optimal performance, this value should not exceed the number of installed or available processors on the PowerCenter Integration Service machine. Valid values are from 1 through 64. Default is 0.
Array Size	No	Real Time	If the Worker Threads value is greater than 0, specifies the size of the storage array, in number of records, for the threads. Valid values are from 1 through 5000. Default is 25.
Map Location	No	Real Time	If you use remote logging to a PowerExchange Logger for Linux, UNIX, and Windows, the location of the source extraction maps. This value must be the node name of the location, as specified in a NODE statement in the PowerExchange dbmover.cfg configuration file on the Integration Service machine. For more information about remote logging, see the <i>PowerExchange CDC Guide for Linux, UNIX, and Windows</i> .
Map Location User	No	Real Time	If you specify a Map Location value, a valid user ID for the map location.
Map Location Password	No	Real Time	If you specify a Map Location value, a valid password for the specified map location user.
CAPL Connection Name Override	No	Real Time	Overrides the default CAPL connection name.

Connection Attribute	Required?	Change or Real Time	Description
Retrieve PWX Log Entries	No	Both	Controls whether PWXPC writes PowerExchange informational messages related to a CDC session in the PowerCenter session log. Default is to write PowerExchange error and warning messages but not informational messages to the session log.

Connection Attribute	Required?	Change or Real Time	Description
PWX Override	No	Both	<p>PowerExchange connection overrides, separated by a semicolon. You can enter one or more of the following optional overrides:</p> <ul style="list-style-type: none"> - APPBUFSIZE=<i>app_buf_size</i>. Specifies the initial size of the application buffer for the connection. Overrides the APPBUFSIZE statement in the DBMOVER configuration file. - CONNECTSTRINGCODEPAGE=<i>code_page</i>. Code page of the characters in the connection string. Enter this override if PowerExchange issues message PWX-07610. - LOWVALUES={Y N}. Indicates whether PowerExchange preserves hexadecimal '0' values, called <i>low values</i>, in source character fields when passing these values to a PowerCenter session. Enter Y to preserve low values. When the session runs, PWXPC can write these values to a VSAM target on z/OS or to a sequential file target on Linux, UNIX, Windows, or z/OS. Overrides the LOWVALUES statement in the DBMOVER configuration file. - PWXNOQUOTES={Y N}. By default, in the SQL SELECT statements that PWXPC uses to select data for session processing, PWXPC adds double-quotation marks (") around every table name and field name that includes spaces or special characters. These special characters are /+ -= ~ ^ & * () [] { } ' ; ? , < > \ \ . If a table is identified by both a schema name and table name in the format <i>schema.table_name</i>, PWXPC places the quotation marks only around the "<i>table_name</i>" value. If for any reason, you cannot use quoted table names or field names in your environment, set PWXNOQUOTES to Y to override the default behavior. - RETLOGINFOMSG={N Y}. By default, PWXPC writes PowerExchange error and warning messages but not informational messages to the session log. If you specify RETLOGINFOMSG=Y, PWXPC writes informational messages as well as error and warning messages to the session log. Note: The RETLOGINFOMSG connection override is equivalent to the Retrieve PWX Log Entries connection attribute. - TCPIP_OP_TIMEOUT=<i>network_operation_timeout</i>. A network operation timeout interval in seconds. For more information about this timeout override, see "Asynchronous Network Communication" on page 127. - TCPIP_CON_TIMEOUT=<i>connection_timeout</i>. A connection timeout interval in seconds. For more information about this timeout override, see "Asynchronous Network Communication" on page 127.
Connection Retry Period	No	Both	<p>The number of seconds the PowerCenter Integration Service attempts to reconnect to the PowerExchange Listener after the initial connection attempt fails. If the Integration Service cannot connect to the PowerExchange Listener within the retry period, the session fails.</p> <p>Default value is 0, which disables connection retries based on this PWXPC connection attribute.</p> <p>Note: PowerExchange Listener connections to source databases other than those used for PowerExchange Express CDC for Oracle do not have connection resilience.</p>

For more information about CDC offload processing and remote logging, see the *PowerExchange CDC Guide for Linux, UNIX, and Windows*.

RELATED TOPICS:

- [“Configuring Encryption and Compression” on page 109](#)
- [“Configuring Pacing” on page 110](#)
- [“Configuring Write Mode” on page 112](#)
- [“Converting Character Data to Strings” on page 110](#)
- [“Retrieving PowerExchange Log Entries” on page 111](#)
- [“Configuring Image Type” on page 113](#)
- [“Configuring the CAPI Connection Name Override” on page 115](#)
- [“Configuring Idle Time” on page 116](#)
- [“Configuring UOW Count” on page 118](#)
- [“Configuring CDC Restart Attributes” on page 117](#)
- [“Configuring PWX Latency and Real-Time Flush Latency” on page 119](#)
- [“Configuring Maximum Rows per Commit” on page 120](#)
- [“Commit Processing for CDC Sessions” on page 157](#)
- [“Managing Session Log and Restart Token File History” on page 181](#)
- [“Configuring Event Table Processing” on page 114](#)
- [“Enabling Recovery for a CDC Session” on page 173](#)

Configuring Connection Attributes

The connection attributes that you use depend on how you want to run the PowerCenter session to extract data from the source. This section highlights some key connection attributes. For a complete list of all attributes that you can configure for a connection type, see the topics for specific connection types.

Use the following key connection attributes, as appropriate:

- Common connection attributes:
 - Compression, Encryption Type, Encryption Level
 - Pacing Size, Interpret as Rows
 - Convert character data to string
 - Retrieve PWX Log Entries
- Batch application and relational connection attributes:
 - Write Mode
- CDC connection attributes:
 - Image Type
 - Event Table
 - CAPI Connection Name Override
 - Idle Time
 - CDC Restart
 - UOW Count

- Real-Time Flush Latency
- Maximum Rows Per commit
- Minimum Rows Per commit
- Asynchronous communication connection attributes:
 - PWX Override: TCPIP network operations timeout
 - PWX Override: TCPIP connection timeout

Common Connection Attributes

The attributes discussed in this section are common to both the PWXPC relational and application connection types.

Configuring Encryption and Compression

When you configure connections to extract or load data with PowerExchange, you can set connection attributes to encrypt or compress data during the PowerCenter session.

Set the **Compression**, **Encryption Type**, and **Encryption Level** connection attributes to compress data or to encrypt data. These attributes override the values in the COMPRESS, ENCRYPT, and ENCRYPTLEVEL statements, if defined, in the DBMOVER configuration file on the Integration Service machine. To enable compression or encryption for a PowerCenter session, be sure to select the appropriate connection attributes.

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 several Informatica products. For more information about implementing SSL authentication, see the "Secure Sockets Layer Support" chapter in the *PowerExchange Reference Manual*.

For more information about implementing SSL authentication in a PowerExchange network, see the *PowerExchange Reference Manual*.

The following table describes the connection attributes for compression and encryption:

Connection Attribute	Required	Description
Compression	No	Select this attribute to compress source data during the PowerCenter session.
Encryption Level	No	<p>If you select AES for Encryption Type, enter one of the following values to indicate the encryption level:</p> <ul style="list-style-type: none"> - 1. Use a 128-bit encryption key. - 2. Use a 192-bit encryption key. - 3. Use a 256-bit encryption key. <p>Default is 1.</p> <p>Note: If you select None for Encryption Type, PowerExchange ignores the Encryption Level value.</p>
Encryption Type	Yes	<p>Select one of the following values for the encryption type:</p> <ul style="list-style-type: none"> - None - AES <p>Default is None.</p> <p>Note: The values of DES and RC2 are deprecated. PowerExchange converts a value of DES and RC2 to AES.</p>

Configuring Pacing

You can configure the pacing size to slow the data transfer rate from the PowerExchange Listener. The pacing size determines the amount of data the PowerExchange Listener passes to the source or target. Configure the pacing size if an external application, database, or the PowerCenter Integration Service node is a bottleneck during the session.

The following table describes the pacing attributes:

Connection Attribute	Required	Description
Pacing Size	No	Enter the amount of data the source system can pass to the PowerExchange Listener. The lower the value, the faster the session performance. Minimum value is 0. Enter 0 for maximum performance. Default is 0.
Interpret as Rows	No	Select to represent the pacing size in number of rows. If you clear this option, the pacing size represents kilobytes. This option is selected by default.

Configuring Timeouts

When you configure connections to extract and load data from PowerExchange, you can configure connection and network operation timeouts.

By default, PowerExchange uses a connection timeout value of 180 seconds, and network operation timeouts are not enabled. Instead, PowerExchange uses heartbeat processing to detect network send and receive timeouts.

You define a connection or network operation timeout value by using the PWX Override connection attribute.

The following table describes the timeout values that you can specify with the PWX Override attribute:

Connection Attribute	Required	Description
PWX Override	No	Specifies PowerExchange timeout values, in seconds. Separate multiple statements with semicolons. - TCPIP_OP_TIMEOUT= <i>network_operation_timeout</i> ; - TCPIP_CON_TIMEOUT= <i>connection_timeout</i> Valid values for TCPIP_OP_TIMEOUT are from 5 to 30758400. Or you can specify 0 or -1 for no timeouts, the default. Valid values for TCPIP_CON_TIMEOUT are from 10 to 2678400. Default is 180.

Converting Character Data to Strings

For some data sources, data fields that have a character datatype such as CHAR or VARCHAR might contain embedded nulls. PowerCenter expects character fields to contain null-terminated strings. If character fields contain embedded nulls, unpredictable results might occur for bulk data movement and CDC sessions.

Usually, this problems occurs for z/OS sources with character fields that intentionally or unintentionally contain embedded nulls.

To avoid this problem, you can convert character fields to string fields. For string data fields, PowerCenter converts each embedded null to a space.

For example, a single character field is a collection of subfields. The user application that writes data to the z/OS source uses null delimiters (x'00') to separate the subfields.

The character field contains the following hexadecimal EBCDIC data:

```
C1C2C300C4C5C6C700C8C9
```

In this data, the double zeroes represent the null delimiters. The user application parses the data into the following hexadecimal strings:

- x'C1C2C3' (translates to text "ABC")
- x'C4C5C6C7' (translates to text "DEFG")
- x'C8C9' (translates to text "HI")

No explicit mapping of the subfields exists. PowerCenter reads only the string "ABC" for the session and truncates the rest of the data, after the first null delimiter.

Besides truncation, other unpredictable results can occur, such as session failure. Problems are less likely to occur if the workflow passes the data directly through to the target without a transformation.

To mitigate these issues, you can convert character data with embedded nulls to strings so that the nulls are handled as spaces. Use one of the following methods:

- To have PowerCenter convert character fields for all sources that use a connection, select the **Convert character data to string** attribute on the connection. By default, this attribute is not selected and PowerCenter passes the nulls in the character fields to the target.
- To manually convert individual character fields in a table, edit the source definition in Source Analyzer. In the **Edit Tables** dialog box, click the **Columns** tab. Then, change the datatype for each CHAR and VARCHAR column that you want to convert to STRING. This method is more efficient if you know which character columns are likely to contain nulls.

If you apply one of these conversion methods to the preceding example EBCDIC data, PWXPC replaces the zeroes with "40" when reading the data:

```
C1C2C340C4C5C6C740C8C9
```

When PowerCenter extracts this data to the target, it replaces the "40" values with spaces:

```
ABC DEFH HI
```

Retrieving PowerExchange Log Entries

PWXPC writes PowerExchange error and warning messages to the PowerCenter session log by default. You can configure PWXPC to also write PowerExchange informational messages to the session log. The session log then contains all of the PowerCenter and PowerExchange messages related to session processing.

PWXPC incorporates PowerExchange messages into PowerCenter message PWXPC_10091 when writing messages to the session log.

To configure PWXPC to include PowerExchange informational messages in the PowerCenter session log, configure one of the following connection attributes in PowerCenter:

- Select the **Retrieve PWX Log Entries** connection attribute. (PWXPC application connection types only)
- Specify the RETLOGINFOMSG=Y option in the **PWX Override** connection attribute. (All PWXPC connection types)

The **Retrieve PWX Log Entries** connection attribute and the RETLOGINFOMSG=Y option of the **PWX Override** connection attribute are equivalent.

If you enable the **Retrieve PWX Log Entries** connection attribute or specify `RETLOGINFOMSG=Y` in the **PWX Override** connection attribute, PWXPC writes all PowerExchange messages, including informational messages, that are related to the session to the PowerCenter session log.

Caution: Informational messages are not limited in size and might take up hundreds of lines in your session log.

Batch Application and Relational Connection Attributes

These attributes are available in both PWXPC batch application and relational connection types unless otherwise noted.

Configuring Write Mode

To configure the write mode, define the **Write Mode** and **Reject File** connection attributes.

Write Mode

Optional. Defines the mode in which to send data to the PowerExchange Listener.

Select one of the following options:

Confirm Write On

Sends data to the PowerExchange Listener and waits for a success or no success response before sending more data. This mode sends data synchronously to the PowerExchange Listener rather than buffering the data. After sending an SQL request, the sender waits for the response from PowerExchange before sending the next SQL request.

This mode provides good error recovery but has slower data transfer rates than the other modes.

To stop session execution when the session encounters non-fatal errors, specify a value greater than 0 in the **Stop on errors** session attribute on the **Config Object** tab of the **Edit Tasks** dialog box.

If you plan to run CDC workflows with the **Resume from last checkpoint** recovery strategy and write data to targets with a PWX batch or relational connection, specify **Confirm Write On** as the write mode for the target connection.

Confirm Write Off

Sends data asynchronously to the PowerExchange Listener by buffering the data. This mode does not wait for a success or no success response. While this mode provides a faster data transfer speed than **Confirm Write On**, it does not provide the ability to determine which SQL statement failed in error situations. If an error occurs, you must reload the entire table if an error occurs to ensure data integrity. Use this setting only if you can reload tables.

Caution: When you use **Confirm Write Off**, PowerCenter statistics are unreliable.

Asynchronous with Fault Tolerance

Combines the speed of **Confirm Write Off** with the error detection of **Confirm Write On**. This mode buffers the data and sends it asynchronously to the PowerExchange Listener. When an SQL error occurs, PowerExchange creates a reject file on the target machine, which contains the data records that the writer could not write to the target. View the file contents to identify and correct the errors without reloading the entire table. You can also specify how to handle specific SQL return codes. To stop session execution when the session encounters non-fatal errors, specify a value greater than 0 in the **Stop on errors** session attribute on the **Config Object** tab of the **Edit Tasks** dialog box.

The **Asynchronous with Fault Tolerance** option is available only for PWX DB2zOS, PWX DB2i5OS, PWX DB2LUW, PWX MSSQLServer, and PWX Oracle relational connections. However, for relational

targets on Linux, UNIX, and Windows, sessions usually load data directory to the targets by using the native drivers instead of a PWX relational connection.

Default is **Confirm Write On**.

Reject File

Optional. A prefix that overrides the default prefix of PWXR for the PowerExchange reject file.

PowerExchange creates the reject file on the target machine when you specify a write mode of **Asynchronous with Fault Tolerance** and an error occurs that prevents data from being written to the target. The reject file contains the rows of data that the writer did not write to the target.

Note: Enter PWXDISABLE to prevent the creation of the reject files.

For more information, see the *PowerExchange Bulk Data Movement Guide*.

CDC Application Connection Attributes

This section describes attributes that are specific to the PWXPC CDC application connection types.

RELATED TOPICS:

- [“Configuring Image Type” on page 113](#)
- [“Configuring Event Table Processing” on page 114](#)
- [“Configuring Idle Time” on page 116](#)
- [“Configuring Maximum Rows per Commit” on page 120](#)
- [“Configuring Minimum Rows per Commit” on page 121](#)
- [“Configuring PWX Latency and Real-Time Flush Latency” on page 119](#)
- [“Configuring the CAPI Connection Name Override” on page 115](#)
- [“Configuring UOW Count” on page 118](#)
- [“Connection Attributes for CDC Commit Processing” on page 122](#)
- [“Extracting CDC Data in Change and Real-time Modes” on page 137](#)

Configuring Image Type

The following table describes the optional **Image Type** connection attribute:

Connection Attribute	Change or Real Time	Description
Image Type	Both	<p>Indicates how PWXPC passes captured updates on a source to CDC sessions that apply the updates to the target.</p> <p>Valid values:</p> <ul style="list-style-type: none">- AI. Process updates as update operations. PWXPC passes each update as a single update record. An update record includes after images of the data only, unless you add before image (BI) and change indicator (CI) fields to the extraction map that is imported to create the source definition for the CDC session.- BA. Process updates as deletes followed by inserts. PWXPC passes each update as a delete record followed by an insert record. The delete record contains the before image of the data, and the insert record contains the after image. <p>Default is BA.</p>

PowerExchange captures before and after image data for update operations on the source, regardless of the source type. The before-image data can always be extracted in real-time mode. However, in batch and continuous extraction modes, only after-image data is available if the changes were processed by PowerExchange Condense or the PowerExchange Logger for Linux, UNIX, and Windows with only after images.

If you specify **BA**, PWXPC generates, for each captured update operation, a delete record that contains the before image of the data and an insert record that contains the after image. If you also define BI and CI fields for some columns in the extraction map that is imported for the source definition, PWXPC populates the BI and CI fields based on the information in the generated delete and insert records.

If you specify **AI**, you can still use before images of the data, if available, in extraction processing. PWXPC can embed before-image data and after-image data in the same update row. For PWXPC to embed before-image data, you must complete the following configuration tasks:

- In the PowerExchange Navigator, add BI and CI fields to the extraction map that you plan to import for the source definition in PowerCenter.
- If you use batch or continuous extraction mode, enter BA for the CAPT_IMAGE parameter in the PowerExchange Condense or PowerExchange Logger for Linux, UNIX, and Windows configuration file. This setting causes both before and after images to be stored in the PowerExchange Condense condense files or PowerExchange Logger log files. When CDC sessions run, they extract data from these files.

Informatica recommends that you use the **AI** setting if you want to process before images of data. CDC sessions can process a single update record more efficiently than separate delete and insert records to get the before image data.

For example, embed before-image data and after-image data in the same update row to handle changes to primary keys. Relational databases that allow changes to primary keys, such as DB2 for z/OS, treat these updates as equivalent to deleting the row and reading it with a new key value. To enable PowerExchange to detect primary key changes, include BI and CI fields for the primary key columns in the extraction map for the source definition. Then, in PowerCenter, define a Flexible Target Key Custom transformation to apply the changes to the target as a delete followed by an insert. Include the transformation in the mapping for the CDC session. If a target relational database does not allow changes to primary keys, updates to primary keys fail.

Note: To use a Flexible Target Key Custom transformation, you must set the **Image Type** attribute to **AI** and configure BI and CI fields in the PowerExchange extraction map for the source.

For more information about adding BI and CI columns to extraction maps, see the *PowerExchange Navigator User Guide*.

RELATED TOPICS:

- [“Configuring Mappings with Flexible Target Key Transformations” on page 184](#)

Configuring Event Table Processing

The following table describes the optional **Event Table** connection attribute:

Connection Attribute	Change or Real Time	Description
Event Table	Real Time	Specifies the PowerExchange extraction map name that is used for event table processing.

You can use event table processing to stop the extraction of changes based on user-defined events, such as an end-of-day event. For example, to stop an extraction process every night, after all of the changes for the

day have been processed, write a change to the event table at midnight. This change triggers PowerExchange to stop reading change data and shut down the extraction process after the current UOW completes.

Event table processing has the following rules and guidelines:

- You can only use event table processing with real-time or continuous extraction modes.
- You must create the event table, and define the applications that can update the table.
- You must register the event table for change data capture from the PowerExchange Navigator.
- The event table and all of the source tables in the CDC session must be of the same source type.

To use event table processing:

1. Create an event table.

The event table must be of the same source type and on the same machine as the change data that is extracted. For example, if you extract DB2 change data on MVS, the event table must be a DB2 table in the same DB2 subsystem as the DB2 source tables for the extraction.

2. In the PowerExchange Navigator, create a capture registration and extraction map for the event table.

When you create a capture registration, the PowerExchange Navigator generates an extraction map.

3. In PowerCenter, create a CDC session, and specify the extraction map name in the **Event Table** attribute on the PWX CDC Real Time application connection.

4. When the defined event occurs, update the event table.

When PowerExchange reads the update to the event table, PowerExchange places an end-of-file (EOF) into the change stream. PWXPC processes the EOF, passes it to the PowerCenter Integration Service, and then shuts down the PowerExchange reader. The PowerCenter Integration Service completes writing all of the data currently in the pipeline to the targets and then ends the CDC session.

Configuring the CAPI Connection Name Override

The following table describes the optional **CAPI Connection Name Override** connection attribute:

Connection Attribute	Change or Real Time	Description
CAPI Connection Name Override	Real Time	Overrides the CAPI connection name from the PowerExchange DBMOVER configuration file.

PowerExchange allows a maximum of eight CAPI_CONNECTION statements in a PowerExchange DBMOVER configuration file. Define multiple CAPI_CONNECTION statements when you want to capture changes from more than one database type through a single PowerExchange Listener on a single machine. For example, you can capture changes for Oracle and DB2 for Linux, UNIX, and Windows sources with a single PowerExchange Listener by specifying source-specific CAPI_CONNECTION statements. You can also specify a default CAPI_CONNECTION statement by adding the CAPI_SRC_DFLT statement in the DBMOVER configuration file. For more information about PowerExchange CAPI_CONNECTION statements, see the *PowerExchange Reference Manual*.

To override the CAPI_CONNECTION statements in the DBMOVER configuration file for a PowerCenter CDC session, specify the **CAPI Connection Name Override** connection attribute.

Configuring Idle Time

The following table describes the optional **Idle Time** connection attribute:

Connection Attribute	Change or Real Time	Description
Idle Time	Real Time	<p>Specifies the number of seconds the PowerExchange Listener remains idle after reaching the end of the change log, as indicated by message PWX-09967, before returning an end-of-file (EOF).</p> <p>Valid values:</p> <ul style="list-style-type: none">-1. The EOF is never returned. The session runs continuously.0. The EOF is returned at the end of log. The session ends successfully.n. The EOF is returned after no data is received for specified number of seconds. The session ends. <p>Default is -1.</p>

Use the **Idle Time** terminating condition to indicate whether a real-time session should run continuously or shut down after a specified period of time.

The **Idle Time** timer starts when the PowerExchange Listener begins reading change data for the sources.

If you enter -1 for **Idle Time**, PowerExchange never returns an EOF to the PowerCenter Integration Server, which causes the session to run continuously. Typically, the default value of -1 is used for real-time sessions.

You can stop continuous extraction sessions by issuing the PowerCenter Workflow Monitor Stop or Abort command, the **pmcmd** commands to stop and abort tasks and workflows, or the PowerExchange STOPTASK command.

- Stopping the session or workflow with the PowerCenter Workflow Monitor or the **pmcmd** stop task commands is a normal termination. PowerCenter performs a graceful stop after the CDC reader and the writers process all of the data in the pipeline and shutdown.
- Aborting the session or workflow with the PowerCenter Workflow Monitor or the **pmcmd abort task** command is an abnormal termination. PowerCenter does not wait for the CDC reader and the writers to shut down or to process all of the data in the pipeline.
- The PowerExchange STOPTASK command stops the extraction task in the PowerExchange Listener and passes an EOF to the Integration Service, which then ends the session successfully.

Warning: In the session properties, verify that the **Commit Type** property specifies **Source** and that the **Commit at End of File** property is disabled. By default, the **Commit at End of File** property is enabled. This property causes data to be committed after the CDC reader commits the restart tokens and shuts down, which results in duplicate data being sent to the targets when the session is restarted.

If you enter 0 for **Idle Time**, PowerExchange returns an EOF to the PowerCenter Integration Service when the end-of-log (EOL) is reached. After the EOF is received, the PowerCenter Integration Service terminates the session successfully, which includes committing all of the data and updating the restart token file. The EOL is determined by what was the current end of the change stream at the point that PowerExchange started to read the change stream. This concept of EOL is required because the change stream is generally not static so the actual EOL is continually moving forward. PowerExchange issues the following message when the EOL is reached:

```
PWX-09967 CAPI i/f: End of log for time yy/mm/dd hh:mm:ss reached
```

For example, if a session starts reading a change stream at 10:00 a.m., the EOL at that point is determined. After PowerExchange reaches that point in the change stream, it will return EOF to the Integration Service. This means that changes recorded in the change stream after 10:00 a.m. will not be processed. Specifying 0

for Idle time is a useful in situations where you want to extract change data for sources periodically as opposed to continuously.

If you enter a positive number for **Idle Time**, the session runs until no data is returned for the period of time specified. After the **Idle Time** limit is reached, PowerExchange send an EOF to the PowerCenter Integration Service and the session terminates successfully. Specifying a low value for **Idle Time**, such as 1, can result in this time being reached before all available data in the change stream has been read.

The following message is issued when the **Idle Time** limit has been reached:

```
[PWXPC_10072] [INFO] [CDCDispatcher] session ended after waiting for [idle_time]
seconds. Idle Time limit is reached
```

This message is also issued when a continuous extraction is stopped with the PowerExchange STOPTASK command. In this case, the *idle_time* variable in the message reports 86400, which is the “never expire” time limit that is used when an **Idle Time** value of -1 is specified.

Tip: On very active systems, a positive value for **Idle Time** might never match. Use 0 if you do not want the session to run continuously.

For example, if you specify an **Idle Time** value of 10 seconds and PowerExchange finds no data for the sources in the change stream for a 10-second period, PowerExchange returns an EOF to the PowerCenter Integration Service, which causes the session to terminate successfully.

If you specify values for **Reader Time Limit** and **Idle Time**, the PowerCenter Integration Service stops reading data from the source when one of these terminating conditions is reached, whichever one is reached first. If the **Reader Time Limit** is reached prior to the **Idle Time** limit, the session stops at that point, even though the **Idle Time** limit has not yet been met.

Warning: Reader Time Limit does not result in normal termination of a CDC session. Use Idle Time instead of Reader Time Limit.

Configuring CDC Restart Attributes

The following table describes the optional restart connection attributes for CDC sessions:

Connection Attribute	Change or Real Time	Description
Application Name	Both	Application name for the extraction. This name is part of the audit trail for data extractions by the PowerExchange Listener. The application name must be unique for each session. The name is case sensitive and cannot exceed 20 characters. Default is the first 20 characters of the workflow name.
RestartToken File Folder	Both	Folder name that contains the restart token override file on the PowerCenter Integration Service node. Default is \$PMRootDir/Restart.
RestartToken File Name	Both	File name in the restart token file folder that contains the restart token override file. PWXPC uses the contents of this file, if any, in conjunction with the state information to determine the restart point for the session. Default is the Application Name value if specified, or the workflow name if Application Name is not specified.

You can use numerous CDC reader application connection attributes to specify restart information. PWXPC uses the restart information to tell PowerExchange the point from which to start reading the captured change data.

Warning: Be careful when you use the default value for **Application Name**. The default might not result in a unique application name. The **Application Name** value and the **RestartToken File Name** values must be unique for every CDC session. If a non-unique name is specified for either attribute, CDC session failures and potential data loss can occur or results are unpredictable.

RELATED TOPICS:

- [“Configuring the Restart Token File” on page 169](#)

Configuring UOW Count

The following table describes the optional **UOW Count** connection attribute:

Connection Attribute	Change or Real Time	Description
UOW Count	Both	The number of units of work (UOWs) that PWXPC processes before it sends a commit request to PowerCenter to commit data to all targets in the CDC session. Valid values: -1. Disables this attribute. 0. Disables this attribute. <i>n</i> . Specifies the number of UOWs that PWXPC processes before sending a commit request. Default is 1.

A unit of work (UOW) is a collection of changes within a single commit scope that a transaction makes on the source system. Each UOW can contain a different number of changes.

PWXPC reads change data from PowerExchange and provides the data to the source qualifier, at which point the count of UOWs begins. When you use a non-zero value for the **UOW Count** attribute, PWXPC issues a real-time flush to commit the change data to the target after the UOW count reaches the specified value. When PWXPC issues a real-time flush, it writes the following message to the session log:

```
[PWXPC_10081] [INFO] [CDCDispatcher] raising real-time flush with restart tokens  
[restart1_token], [restart2_token] because UOW Count [uow_count] is reached
```

For example, if the **UOW Count** value is 10, the PowerCenter Integration Service commits all change data read from the source to the target after the tenth UOW is processed.

The lower you set the **UOW Count** value, the faster the PowerCenter Integration Service commits data to the target. If you require the lowest possible latency for applying change data to targets, specify a **UOW Count** value of 1.

Commit processing is not controlled solely by the **UOW Count** attribute. The **Maximum Rows Per commit**, and **Real-Time Flush Latency** attributes also determine the commit frequency.

If you have many small UOWs, you can use the **UOW Count** or **Minimum Rows Per commit** attribute or both to create a consistent flow of UOWs that can be committed to the target in batches of approximately the same size.

You can also control commits that occur at UOW boundaries based on time by specifying the **Real-time Flush Latency** attribute. Specify the **UOW Count** or **Real-time Flush Latency** attribute or both. If you specify both, a commit occurs when either limit is reached.

Warning: If you specify a low **UOW Count** value, the session might consume more system resources on the target platform because data is committed to the target more frequently. Balance performance and resource consumption with latency requirements when setting the **UOW Count**, **Maximum Rows Per commit**, and **Real-Time Flush Latency** connection attributes.

RELATED TOPICS:

- [“Commit Processing for CDC Sessions” on page 157](#)

Configuring PWX Latency and Real-Time Flush Latency

The following table describes the optional flush latency connection attributes:

Connection Attribute	Change or Real Time	Description
Real-Time Flush Latency in milliseconds	Real Time	The number of milliseconds that must elapse before PWXPC sends a commit request to PowerCenter. When this period elapses, PWXPC continues to read the changes in the current UOW until it reaches the end of the UOW. Then PWXPC sends a commit request to PowerCenter so that the data can be committed to the targets. Valid values: -1. Disables this attribute. 0 to 86400 . Specifies the number of milliseconds that must elapse before PWXPC sends a commit request to PowerCenter. If you specify a value from 0 through 2000, PWXPC uses 2000 milliseconds (2 seconds). Default is 0.
PWX Latency in seconds	Real Time	The maximum number of seconds that PowerExchange on the source machine waits for more change data before it flushes data to PWXPC on the PowerCenter Integration Service machine. Default is 2.

When a CDC session begins, PWXPC passes the **PWX Latency in seconds** attribute value from the connection to the PowerExchange Listener and requests change data from the change stream. The PowerExchange Listener uses this latency value as the Consumer API (CAPI) interface timeout value on the source machine. PowerExchange issues the following message on the source machine to identify the CAPI interface timeout value:

```
PWX-09957 CAPI i/f: Read times out after <number> seconds
```

If you select the **Retrieve PWX Log Entries** attribute on the connection, PWXPC also writes this informational message to the session log.

PowerExchange reads change data into a buffer on the source machine. When the one of the following conditions occur, PowerExchange flushes the buffer that contains the change data to PWXPC on the PowerCenter Integration Service machine:

- The buffer becomes full.
- A commit point occurs.
- The **PWX Latency in seconds** interval, or CAPI interface timeout, expires.

Note: The CAPI interface timeout also affects the speed with which PWXPC processes stop requests from PowerCenter. PWXPC must wait for PowerExchange to return control to it before it can handle a stop request.

After PowerExchange flushes change data to PWXPC, PWXPC provides the change data to the source qualifier and the **Real-Time Flush Latency** interval begins. After the **Real-Time Flush Latency** interval expires

and PWXPC reaches a UOW boundary, PWXPC sends a commit request to PowerCenter and writes the following message to the session log:

```
[PWXPC_10082] [INFO] [CDCDispatcher] raising real-time flush with restart tokens
[restart1_token], [restart2_token] because Real-time Flush Latency [milliseconds]
occurred
```

For example, if the value for real-time flush latency interval is 10 seconds, PWXPC sends a commit request to PowerCenter for all complete UOWs after 10 seconds elapses and the next UOW boundary is met.

The lower you set the **Real-Time Flush Latency** interval, the faster change data is committed to the targets. If you require the lowest possible latency for applying changes to targets, specify a low value for the **Real-Time Flush Latency** interval.

Warning: When you specify low flush latency intervals, the CDC session might consume more system resources on the source, PowerCenter Integration Service, and target systems for the following reasons:

- PowerExchange returns more frequently to PWXPC. As a result, PowerExchange passes fewer rows to PWXPC on each iteration and consumes more resources on the source machine.
- PowerCenter commits change data to the targets more frequently. As a result, more resources are consumed on the PowerCenter Integration Service and target machines.

When you set the flush latency intervals, balance performance and resource consumption with latency requirements.

To control commit processing on the target, you can use the **Real-Time Flush Latency** attribute in conjunction with the **Maximum Rows Per commit** and **UOW Count** connection attributes.

RELATED TOPICS:

- [“Commit Processing for CDC Sessions” on page 157](#)

Configuring Maximum Rows per Commit

The following table describes the optional **Maximum Rows Per commit** connection attribute:

Connection Attribute	Change or Real Time	Description
Maximum Rows Per commit	Both	The maximum number of change records that PWXPC processes before it sends a commit request to PowerCenter to commit data to all targets in the CDC session. PWXPC does not wait for a UOW boundary to commit the change data. Default is 0, which disables this attribute.

You can use the **Maximum Rows Per commit** attribute to commit change records between UOW boundaries. Use this attribute when you have extremely large UOWs in the change stream that might cause locking issues on the target database or high resource consumption on the PowerCenter Integration Service machine. PWXPC uses the **Maximum Rows Per commit** value to commit data to the targets before the end of a UOW is received, a process called a *subpacket commit*.

Important: Because PWXPC can commit change data to targets between UOW boundaries, referential integrity (RI) might be compromised. Do not use this connection attribute if you have targets in the CDC session with RI constraints.

The **Maximum Rows Per commit** value is the number of change records in a source UOW that PWXPC processes before issuing a real-time flush to commit the change data to the target. When PWXPC flushes change data to the target, it writes the following message to the session log:

```
[PWXPC_12128] [INFO] [CDCDispatcher] raising real-time flush with restart tokens  
[restart1_token], [restart2_token] because Maximum Rows Per commit [commit_count]  
occurred
```

For example, if each UOW contains 1,000 change records and you specify 100 for **Maximum Rows Per commit**, PWXPC issues a real-time flush to commit the change records after each 100 records, which results in 10 commits for each UOW.

As part of the commit processing, any locks in the target databases for these changes are released.

PWXPC resets the **Maximum Rows Per commit** counter when either the specified number of records is met or the end of the UOW is reached.

Commit processing is not controlled solely by the **Maximum Rows Per commit** attribute. The **UOW Count** and **Real-Time Flush Latency** attributes also determine the target commit frequency. The **Maximum Rows Per commit** attribute differs from the **UOW Count** attribute in that it specifies a count of records within a UOW whereas the **UOW Count** specifies a count of complete UOWs.

By using a subpacket commit for large UOWs, you can minimize lock contention on the target and reduce storage use on the PowerCenter Integration Service machine. However, if you specify a low **Maximum Rows Per commit** value, the session might consume more system resources on the target machine because PWXPC sends commit requests to the target more frequently. Balance performance and resource consumption with latency requirements when setting the **Maximum Rows Per commit**, **UOW Count**, and **Real-Time Flush Latency** values.

When a session processes changes for multiple sources, the **Maximum Rows Per commit** count is cumulative across all sources in the group. PWXPC sends a commit request to PowerCenter when the maximum count of change records is reached, regardless of the number of sources to which the changes apply. For example, assume that a UOW contains 900 change records for one source followed by 100 change records for a second source and then another 500 change records for the first source. If you set the **Maximum Rows Per commit** attribute to 1000, PWXPC issues the commit after the thousandth change record, that is, after the 100 changes for the second source.

Warning: If a UOW contains changes for multiple source tables, the **Maximum Rows Per commit** attribute might cause commits to be generated at points in the change stream where the relationship linking the source tables is inconsistent. In this situation, target commit failures can occur.

RELATED TOPICS:

- [“Commit Processing for CDC Sessions” on page 157](#)

Configuring Minimum Rows per Commit

The following table describes the optional **Minimum Rows Per commit** connection attribute:

Connection Attribute	Change or Real Time	Description
Minimum Rows Per commit	Real Time	The minimum number of change records that the PowerExchange Listener must read from the change stream before it passes a commit record to PWXPC. Before reaching this minimum, the PowerExchange Listener passes only change records to PWXPC and discards any commit records. Default is 0, which disables this attribute.

If the change stream contains many small UOWs, you can use the **Minimum Rows Per commit** attribute to create larger UOWs of a more uniform size.

This attribute specifies the minimum number of change records that the PowerExchange Listener must read from the change stream before it passes a commit request to PWXPC. Until the minimum rows limit is met, the PowerExchange Listener discards any commit records that it reads from the change stream and passes only change records to PWXPC. After the minimum rows limit is met, PowerExchange passes the next commit record to PWXPC and then resets the minimum rows counter.

Online transactions that run in transaction control systems such as CICS and IMS often issue a commit after making only a few changes, which results in many, small UOWs in the change stream. PowerExchange and PWXPC can process fewer, larger UOWs more efficiently. Therefore, by using this attribute to increase the size of UOWs, you can improve CDC processing efficiency.

A minimum rows limit does not damage the referential integrity of the change data because PowerExchange does not create new commit points in the change stream. PowerExchange only skips some of the original commit records in the change stream.

Connection Attributes for CDC Commit Processing

The following table describes the optional CDC connection attributes that you can use to control target commit processing:

Connection Attribute	Real Time or Change	Description
Maximum Rows Per commit	Both	The maximum number of change records that PWXPC processes before it sends a commit request to PowerCenter to commit data to all targets in the CDC session. PWXPC does not wait for a UOW boundary to commit the change data. Default is 0, which disables this attribute.
Minimum Rows Per commit	Real Time	The minimum number of change records that PowerExchange must read from the change stream before it can pass any source commit record to PWXPC. Before reaching this minimum, PowerExchange passes only the change records, without any commit records, to PWXPC. Default is 0, which disables this attribute.

Connection Attribute	Real Time or Change	Description
Real-time Flush Latency in milliseconds	Real Time	<p>The number of milliseconds that must elapse before PWXPC sends a commit request to PowerCenter. When this period elapses, PWXPC continues to read the changes in the current UOW until it reaches the end of the UOW. Then PWXPC sends a commit request to PowerCenter so that the data can be committed to the targets.</p> <p>Valid values:</p> <p>-1. Disables this attribute.</p> <p>0 to 86400. Specifies the number of milliseconds that must elapse before PWXPC sends a commit request to PowerCenter. If you specify a value from 0 through 2000, PWXPC uses 2000 milliseconds (2 seconds).</p> <p>Default is 0.</p>
UOW Count	Both	<p>The number of UOWs that PWXPC processes before it sends a commit request to PowerCenter to commit data to all targets in the CDC session.</p> <p>Valid values:</p> <p>-1. Disables this attribute.</p> <p>0. Disables this attribute.</p> <p>n. Specifies the number of UOWs that PWXPC processes before sending a commit request.</p> <p>Default is 1.</p>

You can specify values for all of these commit control attributes. However, PWXPC flushes the data buffer to commit change data to the targets only when one of the following attribute values is met, whichever is first:

- **Maximum Rows Per commit**
- **Real-time Flush Latency**
- **UOW Count**

After PWXPC commits the change data, it resets the **Maximum Rows Per commit**, **Real-time Flush Latency**, and **UOW Count** counters and continues to read change data. When one of these commit-control attribute values is met again, PWXPC commits the new change data to the targets. Commit processing continues until the CDC session is stopped, ends normally, or ends abnormally. When the PWXPC CDC reader ends normally, PWXPC issues a final commit to flush all of the complete, buffered UOWs and their final restart tokens to the targets. Before ending, the PWXPC CDC reader writes the following message to the session log:

```
PWXPC_12075 [INFO] [CDCRestart] Session complete. Next session will restart at: Restart 1 [restart1_token] : Restart 2 [restart2_token]
```

Restriction: If you select the **Commit On End Of File** property on the session **Properties** tab, duplicate data might occur on the targets because the PowerCenter Integration Service commits any remaining change data in the buffer to the targets. This final commit occurs after the PWXPC CDC reader has committed all complete UOWs in the buffer, along with their restart tokens, to the targets. As a result, the restart tokens might represent a point in the change stream that is earlier than final change data that the PowerCenter Integration Service commits to the targets. To prevent possible duplicate data when you restart CDC sessions, set the **Commit Type** session property to **Source** and clear the **Commit On End Of File** property.

RELATED TOPICS:

- [“Commit Processing for CDC Sessions” on page 157](#)

Target Latency

Target latency is the total time that PWXPC takes to extract change data from the change stream and that the PowerCenter Integration Service takes to apply that data to the targets. If this processing occurs quickly, target latency is low.

The values that you use for the commit control attributes affect target latency. You must balance target latency requirements with resource consumption on the PowerCenter Integration Service machine and the target databases.

Lower target latency results in higher resource consumption because the PowerCenter Integration Service must flush the change data more frequently and the target databases must process more commit requests.

The following default values for commit control attributes result in the lowest latency:

- 0 for **Maximum Rows Per commit**, which disables this option
- 0 for **Minimum Rows Per commit**, which disables this option
- 0 for **Real-time Flush Latency in milli-seconds**, which is equivalent to 2000 milliseconds or 2 seconds
- 1 for **UOW Count**

These values can decrease target latency because PWXPC commits changes after each UOW or on UOW boundaries. However, these values also cause the highest resource consumption on the source system, the Integration Service machine, and the target databases. Or these values might decrease throughput because change data flushes too frequently for the PowerCenter Integration Service or the target databases to handle.

To lower resource consumption and potentially increase throughput for CDC sessions, specify a value greater than the default value for only one of the following attributes:

- **Minimum Rows Per commit**
- **Real-time Flush Latency in milli-seconds**
- **UOW Count**

Disable the unused attributes.

Examples of Using CDC Commit Processing Attributes

The following examples show how the commit control attributes affect CDC commit processing with PWXPC.

Example 1. Subpacket Commit and UOW Count

This example uses the **Maximum Rows Per commit** and **UOW Count** attributes to control target commit processing. The change data consists of UOWs of the same size. Each UOW contains 1,000 change records. The commit control attributes have the following values:

- **Maximum Rows Per commit** is set to 300.
- **Minimum Rows Per commit** is set to 0, which disables this attribute.
- **Real-time Flush Latency in milli-seconds** is set to 0, which causes 2 seconds to be used.
- **UOW Count** is set to 1.

Based on the maximum rows value, PWXPC flushes the data buffer after reading the first 300 records in a UOW. This action commits the change data to the targets. PWXPC continues to commit change data to the targets every 300 records.

PWXPC commits change data on UOW boundaries only for the UOW count and real-time flush latency interval. If the real-time flush latency interval expires before PWXPC reads 300 change records, PWXPC still commits change data based on the maximum rows value because that threshold is met before a UOW boundary occurs.

When the end of the UOW is read, PWXPC commits the change data because the **UOW Count** value is 1. PWXPC resets the UOW and maximum rows counters and the real-time flush latency timer each time PWXPC issues a commit. Because all of the UOWs have the same number of change records, PWXPC continues to read change data and commit change data to the targets at the same points in each UOW.

In this example, PWXPC commits change data at the following points:

- 300 change records based on the maximum rows value
- 600 change records based on the maximum rows value
- 900 change records based on the maximum rows value
- 1,000 change records based on the UOW count value

Example 2. UOW Count and Time-Based Commits

This example uses the **UOW Count** and **Real-time Flush Latency in milli-seconds** attributes to control commit processing. The change data consists of UOWs of varying sizes. The commit control attributes have the following values:

- **Maximum Rows Per commit** is set to 0, which disables this attribute.
- **Minimum Rows Per commit** is set to 0, which disables this attribute.
- **Real-time Flush Latency in milli-seconds** is set to 5000, which is equivalent to 5 seconds.
- **UOW Count** is set to 1000.

Initially, PWXPC reads 900 complete UOWs in 5 seconds. At that point, the real-time flush latency interval has expired. As a result, PWXPC flushes the data buffer to commit change data to the targets. PWXPC then resets both the UOW counter and real-time flush latency timer. When PWXPC reaches the thousandth UOW, PWXPC does not commit change data to the targets because the UOW counter was reset to 0 after the last commit.

PWXPC reads the next 1,000 UOWs in 4 seconds, which is less than the real-time flush latency timer. PWXPC commits this change data to the target because the UOW counter has been met. PWXPC then resets the real-time flush latency timer and the UOW counter.

PWXPC continues to read change data and commit change data to the targets based on the UOW count or real-time flush latency limit, whichever is met first.

In this example, PWXPC commits change data at the following points:

- After UOW 900 because the real-time latency flush latency timer was met first.
- After UOW 1,900 because the UOW count was met first during the second commit cycle.

Example 3. Minimum Rows and UOW Count

This example uses the **Minimum Rows Per commit** and **UOW Count** attributes to control target commit processing. The change data consists of UOWs of the same size. Each UOW contains ten change records. The commit control attributes have the following values:

- **Maximum Rows Per commit** is set to 0, which disables this attribute.
- **Minimum Rows Per commit** is set to 100.
- **Real-time Flush Latency in milli-seconds** is set to -1, which disables this attribute.

- **UOW Count** is set to 10.

PWXPC passes the minimum number of rows value to PowerExchange and requests change data from the change stream. Because the minimum rows value is 100, PowerExchange skips commit records in the first 9 UOWs. When PowerExchange reads the last change record in the tenth UOW, the minimum rows limit is met. PowerExchange then passes the commit record for the tenth UOW to PWXPC and resets the minimum rows counter. PWXPC increases the UOW counter to 1.

PowerExchange and PWXPC continue to read the change data until the UOW counter is 10. At this point, PWXPC flushes the data buffer to commit change data to the targets and resets the UOW counter.

In this example, PWXPC commits change data after 1,000 change records. This frequency is equivalent to after every 10 UOWs because each UOW contains 100 change records and the **UOW Count** is 10.

Connection Attributes for Multiple-Record Writes

When you create a connection for a source or target in a workflow that performs multiple-record writes, observe the following rules and guidelines:

- You can use multiple-record writes with PWXPC connections only. Multiple-record writes with PowerExchange ODBC connections are not supported.
- Select **Filter After** for the **Offload Processing** source and target connection attribute. PowerExchange and PWXPC perform offload processing on the Integration Service machine.

If you select a different value for **Offload Processing**, PowerExchange changes the value to **Filter After**.

However, if RBAs are being read using the `GetDatabaseKey` or `GetIMSRBAByLevel` function from an IMS unload file in the BMC LONG format, you must select **No** for the **Offload Processing** source connection attributes. Otherwise, the session will fail with message PWX-03803.

- Multithreaded processing is supported for IMS source connections if you select **Filter After** for the **Offload Processing** connection attribute and set the **Worker Threads** connection attribute to a non-zero value.
 - Multithreaded mapping processing is not supported for IMS target connections. If you set the **Worker Threads** connection attribute to a non-zero value, the session ignores this setting.
 - Multithreaded processing is not supported for sequential or VSAM connections.
 - Multithreaded processing is not supported for sequential or VSAM source or target connections.
- If you set the **Worker Threads** connection attribute to a nonzero value, the setting is ignored.
- Select **Off** for the **Confirm Write** source and target connection properties. If you select a different value, PowerExchange changes the value to **Off**.
 - In certain cases, you might need to change the value of the **CSQ_MEMSIZE** parameter of the **PWX Override** connection property. This property defines the maximum amount of memory that the cache can consume for multiple-record writes.

RELATED TOPICS:

- [“Multiple-Record Writes to Nonrelational Targets” on page 50](#)
- [“Considerations for Sessions that Perform Multiple-Record Writes” on page 149](#)

Asynchronous Network Communication

PowerExchange uses asynchronous network communication for all send and receive data operations between a PowerExchange client and a PowerExchange Listener. With asynchronous communication, PowerExchange uses separate threads for network processing and data processing, so that network processing overlaps with data processing.

To detect unsuccessful connection attempts or failed connections, PowerExchange provides several types of timeout processing, including heartbeat timeout processing. During periods of inactivity, the network threads on the PowerExchange client and PowerExchange Listener send and receive heartbeat data. If no heartbeat or non-heartbeat data is sent or received during the heartbeat timeout interval, PowerExchange terminates the connection and issues an error message.

PowerExchange asynchronous communication, including heartbeat processing, provides the following benefits:

- Improved session performance for streaming data transfers
- Early detection of connection failure situations
- Timeout processing that does not require configuration in most cases

Asynchronous Read and Write Processing

Asynchronous read and write operations use application threads, network threads, and circular buffers to exchange data between a PowerExchange client and PowerExchange Listener.

The following steps summarize the asynchronous read process:

1. The application thread on the PowerExchange Listener writes data to the circular buffer on the PowerExchange Listener.
2. The network thread on the PowerExchange Listener reads data from the circular buffer and sends the data across the network.
3. The network thread on the PowerExchange client reads data from the network and writes the data to the circular buffer on the PowerExchange client.
4. The application thread on the PowerExchange client reads the data from the circular buffer and processes the data.

The following steps summarize the asynchronous write process:

1. The application thread on the PowerExchange client writes data to the circular buffer on the PowerExchange client.
2. The network thread on the PowerExchange client reads data from the circular buffer and sends the data across the network.
3. The network thread on the PowerExchange Listener reads data from the network and writes the data to the circular buffer on the PowerExchange Listener.
4. The application thread on the PowerExchange Listener reads the data from the circular buffer and writes the data to the target.

Timeout Processing

To detect unsuccessful connection attempts or failed connections, PowerExchange provides the following types of timeouts:

Connection timeouts

Connection timeouts are used to detect an unsuccessful connection attempt. By default, PowerExchange tries to connect for 180 seconds. If a connection cannot be established, PowerExchange issues an error message. You can change the connection timeout interval by using the PWX Override connection attribute.

Heartbeat timeouts

During periods of inactivity, the network threads on the PowerCenter client and PowerCenter Listener send and receive heartbeat data. If no data, including heartbeat data, is sent or received over an interval of 500 seconds, PowerExchange terminates the connection and issues an error message.

Network operation timeouts

You can optionally configure network operation timeouts. When the client-side application thread perceives a single network operation to exceed the specified number of seconds, PowerExchange ends the connection and issues a timeout error message.

By default, network operation timeouts are not enabled, and PowerExchange uses heartbeat processing only to detect send and receive timeouts.

Note: The TIMEOUTS statement in the DBMOVER configuration file is deprecated and no longer has any effect.

Configuring Connection and Network Operation Timeouts

Use this procedure to override the default connection timeout or to enable a network operations timeout.

To configure connection and network operation timeouts:

1. In the Workflow Manager, connect to a PowerCenter repository.
2. Complete one of the following actions:
 - Click **Connections > Relational**.
 - Click **Connections > Application**.

The **Relational Connection Browser** dialog box or **Application Connection Browser** dialog box appears.

3. In the **Select Type** field, select the type of connection for which you want to configure timeouts.
4. Click **Edit** or **New**, based on whether you are creating or editing a connection.

The **Connection Object Definition** dialog box appears.

5. Scroll to the **PWX Override** connection attribute. In the **Value** column, specify the number of seconds for the connection timeout interval, network operation timeout interval, or both.

If you specify both overrides, in either order, separate them with a semicolon. Use the following syntax:

```
TCPIP_OP_TIMEOUT=network_operation_timeout;TCPIP_CON_TIMEOUT=connection_timeout
```

6. Click **OK**.

CHAPTER 5

Working with Sessions

This chapter includes the following topics:

- [Working with Sessions Overview, 129](#)
- [Extracting Data in Batch Mode, 129](#)
- [Extracting CDC Data in Change and Real-time Modes, 137](#)
- [Loading Data to PowerExchange Targets, 143](#)
- [Considerations for Sessions that Perform Multiple-Record Writes, 149](#)
- [Constraint-Based Loading for Relational Targets, 153](#)
- [Variables for Session and Connection Properties, 154](#)
- [Workflows that Connect to a PowerExchange Listener Service, 155](#)
- [Configuring PowerExchange Services for Workflows, 156](#)
- [Pipeline Partitioning, 156](#)
- [Commit Processing for Bulk Data Movement Sessions, 156](#)
- [Commit Processing for CDC Sessions, 157](#)
- [Stored Procedure Calls in Pre- and Post-Session SQL Commands, 157](#)
- [Preserving Low Values in Source Character Fields, 158](#)

Working with Sessions Overview

After you create mappings in the PowerCenter Designer, you can create a session and use the session in a workflow to extract, transform, and load data. You create sessions and workflows in the Workflow Manager.

You can create a session in a workflow to extract data in batch, change, or real-time mode. You determine how you want the PowerCenter Integration Service to extract the data when you configure the session. You can also create a session to load data to a target.

Extracting Data in Batch Mode

To extract data in batch mode, you must select a connection for batch mode and configure session properties. The connection you select and the session properties you configure differ depending on the type of source data.

Data Extraction from Multiple Sequential or VSAM ESDS Files

You can extract data from multiple sequential or VSAM ESDS files with the same record layout by using PowerExchange file list processing.

You must use a filelist file. A *filelist file* specifies the names of the files from which source data is to be extracted. The PowerExchange Listener extracts data from all of the files named in the filelist file in the order in which the files are listed.

To request file list processing for a sequential or VSAM ESDS source, configure the following session attributes:

- Specify the filelist file name in the **File Name Override** attribute.
- Select the **Filelist File** option.

RELATED TOPICS:

- [“Configuring Batch Mode Sessions for Nonrelational Sources” on page 131](#)

Overrides for the IMS Access Method and Related Properties

You can override the access method for a PowerExchange IMS source or target in the session properties. You can also override related session properties, such as IMS SSID, PSB name, PCB name, and PCB number depending on the access type.

Use overrides for the following reasons:

- Gain greater flexibility in configuring bulk data movement sessions. When you are ready to run a session, you can switch between the IMS ODBA and DL/I BATCH access methods as needed. Use ODBA to run small, frequent database queries as PowerExchange Listener subtasks to read or write data. Run DL/I batch or BMP jobs at night to process mass updates that require longer access to the database.
- Create fewer PowerExchange data maps. You do not need to create a duplicate data map to change the access method or a related property. With fewer data maps, you also can create fewer PowerCenter mappings and workflows.
- Avoid creating very large PSBs that include many PCBs and require large buffers. Large PSBs are sometimes used to avoid exceeding the maximum of ten NETPORT statements in the DBMOVER member. Instead, you can override the PSB and PCB number at session run time.
- Use one pair of NETPORT and LISTENER statements for DL/I or BMP access to multiple PSBs.
- Override the IMS SSID in an ODBA data map to point to a source or target on another IMS subsystem. You can also use an IMS SSID override to direct a BMP netport job to another IMS SSID or to use another DBD library for an IMS source unload file.

In PowerCenter, enter the session overrides on the **Mapping** tab of the **Edit Tasks** dialog box. The following overrides are available for the source and target, depending on the access method:

- **IMS AM Override.** Overrides the access method.
- **IMS SSID Override.** Overrides the IMS SSID.
- **IMS PSBNAME Override.** Overrides the PSB name. Available for IMS ODBA and DL/I batch access.
- **IMS PCBNAME Override.** Overrides the PCB name. Available for IMS ODBA access only.
- **IMS PCBNUMBER Override.** Overrides the PCB number. Available for DL/I batch access only.

Note: In the PowerExchange Navigator, you can override these properties for a database row test of an IMS data map.

Configuring Batch Mode Sessions for Nonrelational Sources

When you configure a session to extract data from a nonrelational source in batch mode, you can override data map information for the source. PWXPC and PowerExchange use this information to extract the source data.

1. In the Task Developer, double-click a session with a nonrelational source.
2. On the **Mapping** tab, expand the **Sources** node in the **Transformations** tree. Then click a nonrelational source.

Under **Readers**, PowerCenter sets the PowerExchange batch reader for the source type. For nonrelational batch sources, the reader name has one of the following formats:

```
PowerExchange Batch Reader for database_type  
PowerExchange Reader for database_type
```

The *database_type* variable represents one of the following values:

- ADABAS
- ADABAS Unload Files
- DB2 Datamaps
- DB2 Unload Datasets
- DATACOM
- IDMS
- IMS
- Sequential Files
- VSAM Files

Note: The name of the reader cannot be altered. The only exception to this rule is for Adabas. You can use either ADABAS or ADABAS Unload Files.

3. Under **Connections**, in the **Value** field, select a PWX NRDB Batch connection.
4. Under **Properties**, set the session attributes that apply to the source type, as needed.

The following table describes these attributes:

Attribute Name	Description
Schema Name Override	Overrides the schema name in the source PowerExchange data map.
Map Name Override	Overrides the data map name of the source PowerExchange data map.
File Name	For the ADABAS Unload source type, specifies the file name of the unloaded Adabas database. Required for the ADABAS Unload source type.
ADABAS Password	For the ADABAS source type, the database password. If the ADABAS FDT for the source file is password-protected, enter the ADABAS FDT password.

Attribute Name	Description
Database Id Override	For the ADABAS and ADABAS Unload source types, overrides the ADABAS Database ID in the PowerExchange data map.
File Id Override	For the ADABAS and ADABAS Unload source types, overrides the Adabas file ID in the PowerExchange data map.
DB2 Sub System Id	For the DB2 Datamaps source type, overrides the DB2 subsystem ID in the PowerExchange data map.
DB2 Table name	For the DB2 Datamaps source type, overrides the DB2 table name in the PowerExchange data map.
Unload File Name	For the DB2 Unload Datasets source type, overrides the DB2 unload file name in the PowerExchange data map.
Filter Overrides	<p>Filters the source data that PowerExchange reads based on specific conditions that you define.</p> <p>PWXPC adds the filter conditions in a WHERE clause on a SELECT SQL statement and then passes the SQL statement to PowerExchange for processing. You can use any filter condition syntax that PowerExchange supports for NRDB SQL.</p> <p>For a single-record source, use the following syntax:</p> <pre>filter_condition</pre> <p>For example, the following filter condition selects records where a column called TYPE has a value of A or D:</p> <pre>TYPE='A' or TYPE='D'</pre> <p>For a multiple-record source, use one of the following syntax alternatives:</p> <pre>filter_condition</pre> <pre>group_name1=filter; group_name2=filter;...</pre> <p>The <i>group_name</i> syntax limits the SQL query condition to a specific record in a multi-record source definition. If you do not use the <i>group_name</i> syntax, the SQL query condition applies to all records in the multi-record source definition.</p> <p>For example, to select only records that contain an ID column value of "DBA" for a multi-record source that has USER1 and USER2 records, specify one of the following SQL query conditions:</p> <pre>USER1=ID='DBA';USER2=ID='DBA'</pre> <pre>ID='DBA'</pre>
IMS Unload File Name	For the IMS source type, an IMS database unload file name. Required if you want to read source data from the backup file instead of from the IMS database. For a multiple-record write to an IMS unload file, required for both the source and target.

Attribute Name	Description
IMS AM Override	<p>For the IMS source type, overrides the IMS access method in the imported data map for the source with the other available access method. The session then uses the override access method at run time.</p> <ul style="list-style-type: none"> - If you imported a source data map that specifies the DL/1 BATCH access method, enter O to override it with the IMS ODBA access method. For ODBA access, you must also specify the IMS PSBNAME Override and IMS PCBNAME Override attributes. - If you imported a source data map that specifies the IMS ODBA access method, enter D to override it with the DL/1 BATCH access method, which provides DL/I or BMP access. You must also specify the IMS PCBNUMBER Override attribute. <p>Important: Before you run the session with an access method override, ensure that you complete the PowerExchange configuration tasks for the new access method. For example, if the override is DL/1 BATCH, you must configure LISTENER and NETPORT statements in the DBMOVER member and configure the netport JCL. If the override is IMS ODBA, you must perform other configuration tasks. For more information, see "IMS Data Maps" in the <i>PowerExchange Navigator User Guide</i>.</p>
IMS SSID Override	<p>For the IMS source type, if you imported an IMS ODBA data map for the source and did not override the access method, use this attribute to override the IMS subsystem ID (SSID) from the data map for the session. If you specified ODBA access as an override in the IMS AM Override session attribute, you must enter this value. An SSID is required for ODBA access.</p> <p>If the session has an IMS unload file source, you can use this override to point to another IMSID statement in the DBMOVER member for the purpose of changing from one DBD library to another DBD library. By using the override, you can switch DBD libraries without editing or adding any IMSID statement and restarting the PowerExchange Listener. For example, use this override to test changes that you made to a DBD library against an unload file.</p> <p>If you use a netport job with BMP access to IMS, you can use this override with the %IMSID substitution variable in the netport JCL to specify an IMS SSID to use for the session. This override replaces the substitution variable. By using the override with the substitution variable, you can use the same netport JCL to access multiple IMS environments, such as development, test, and production environments.</p> <p>Note: An IMS SSID is not required for DL/I batch access to IMS data or for access to an IMS unload file.</p>
IMS PSBNAME Override	<p>For the IMS source type, if you imported an IMS ODBA data map for the source and did not override the access method, this value overrides the PSB name from the data map. If you specified ODBA access as an override in the IMS AM Override attribute, you must enter this value. A PSB name is required for ODBA access.</p> <p>If you use DL/I batch or BMP access and specify this override, you must also specify the PSB=%PSBNAME substitution variable in the netport JCL. The override value then replaces the substitution variable in the JCL.</p> <p>If you specify the PSB=%1 substitution variable instead of PSB=%PSBNAME in the netport JCL, the session uses the PSB name from the NETPORT statement, if specified. In this case, you need a separate NETPORT statement for each PSB. To avoid exceeding the limit of ten NETPORT statements in the DBMOVER member, use this override with %PSBNAME substitution variable instead.</p> <p>Note: A PSB name is not used for access to an IMS source unload file.</p>
IMS PCBNAME Override	<p>For the IMS source type, if you imported an IMS ODBA data map for the source and did not override the access method, this value overrides the PCB name from the data map. If you specified ODBA access as an override in the IMS AM Override attribute, you must enter this value. A PCB name is required for ODBA access.</p> <p>A PCB name is not used for DL/I batch or BMP access or for access to an IMS unload file.</p>

Attribute Name	Description
IMS PCBNUMBER Override	<p>For the IMS source type, if you imported a DL/1 BATCH data map for the source and did not override the access method, this value overrides the PCB number from the data map. If you specified DL/I access as an override in the IMS AM Override attribute, you must enter this value. A PCB number is required for DL/I or BMP access.</p> <p>A PCB number is not used for IMS ODBA access or for access to an IMS unload file.</p>
File Name Override	<p>For the VSAM Files and Sequential Files source types, overrides the data set or file name in the PowerExchange data map.</p> <p>Enter the complete data set or file name.</p> <p>For the i5/OS, the format is: <i>library_name/file_name</i>.</p> <p>If you select the Filelist File check box, enter the name of a filelist file in this attribute. A filelist file is a list of files.</p>
Filelist File	<p>For the VSAM Files and Sequential Files source types, identifies the file that contains a list of files. Select this attribute only if you entered a filelist file in the File Name Override field.</p>
SQL Query Override	<p>Overrides the SQL query sent to PowerExchange, including any filter overrides.</p> <p>PWXPC replaces the default SQL query with the SQL statement that you enter and passes the SQL statement to PowerExchange for processing. You can enter any SQL statement that PowerExchange supports for NRDB SQL.</p> <p>For example, you can select records from table USER where a column called TYPE has a value of A or D by specifying the following SQL query override:</p> <pre>Select ID, NAME from USER where TYPE='A' or TYPE='D';</pre> <p>For a multiple-record source, use the following syntax:</p> <pre>group_name1=sql_query_override1; group_name2=sql_query_override2;...</pre> <p>For example, you can select only records with ID column values that contain DBA for a multi-record source with two records called USER1 and USER2 by specifying the following SQL query override:</p> <pre>USER1=Select ID, NAME from USER1 where ID='DBA'; USER2=Select ID, NAME from USER2 where ID='DBA';</pre>

Attribute Name	Description
PWX Partition Strategy	<p>For offloaded DB2 Unload, VSAM Files, and Sequential Files source types, specifies one of the following partitioning strategies:</p> <ul style="list-style-type: none"> - Single Connection. PowerExchange creates a single connection to the data source. Any overrides specified for the first partition are used for all partitions. With this option, if you specify any overrides for other partitions that differ from the overrides for the first partition, the session fails with an error message. - Overrides Driven. If the specified overrides are the same for all partitions, PowerExchange creates a single connection to the data source. If the overrides are not identical for all partitions, PowerExchange creates multiple connections.
Flush After N Blocks	<p>For multiple-record sources, specifies the maximum number of block flushes that can occur without any one block being flushed.</p> <p>For bulk multiple-record sources, by default, PWXPC flushes blocks of data only when the buffers are completely full or at end-of-file. If some record types do not have as much data as others, flushing might not occur often. In this case, the record types might not have data on the target for a long time, thereby blocking flushes on the writer side.</p> <p>To ensure that buffers for all record types are flushed at a regular interval, define this Flush After N Blocks session property. This property specifies the maximum number of block flushes that can occur across all record types without any one block being flushed. A value of zero disables this feature and causes flushing to occur only when blocks are full.</p> <p>Valid values for the property are -1 to 100000.</p> <p>The default value of -1 works in the following manner:</p> <ul style="list-style-type: none"> - For all multiple-record sources that do not use sequence fields, process the same as Flush After N Blocks = 0, which disables this feature and flushes only when blocks are full . - For all multiple-record sources that use sequence fields, use Flush After N Blocks = 7 * (number of record types in the source).

5. Click **OK**.

RELATED TOPICS:

- [“Filtering Source Data with PWXPC” on page 205](#)

Configuring Session Properties for Relational Batch Mode Sessions

To extract data from a relational source in batch mode, select a PWXPC relational database connection. You can then configure the session properties.

Session properties unique for PowerExchange are noted.

To configure session properties for a relational batch mode session:

1. In the Task Developer, double-click a session with a relational source to open the session properties.
2. Click the **Sources** view on the **Mapping** tab.
3. In the **Reader** field of the **Readers** settings, select **Relational Reader**. For DB2 for z/OS, you can also select a PowerExchange Reader for DB2 Image Copy reader.
4. If you use the relational reader, select one of the following relational database connections:
 - For DB2 for i5/OS, select PWX DB2i5OS.
 - For DB2 for Linux, UNIX, and Windows, select PWX DB2LUW for DB2.
 - For DB2 for z/OS, select PWX DB2zOS.

- For Microsoft SQL Server, select PWX MSSQLServer.
- For Oracle, select PWX Oracle.

If you use the DB2 Image Copy reader, select a PWX NRDB Batch application connection.

5. Optionally, click the **Sources** view on the **Mapping** tab, and set attributes in the **Properties** area.

The following table describes the attributes that you can configure:

Attribute Name	Description
Schema Name Override	Overrides the source schema name.
Map Name Override	Overrides the source table name.
DB2 Sub System Id	Overrides the DB2 instance name in the PowerExchange data map.
Image Copy Dataset	For DB2 image copy sources, provides the image copy data set name. If not specified and the table is in a non-partitioned table space, the most current image copy data set with TYPE=FULL and SHRLEVEL=REFERENCE is used. If the table is in a partitioned table space, you must specify the Image Copy Dataset attribute.
Disable Consistency Checking	<p>If cleared for a DB2 image copy source, PowerExchange reads the catalog to verify that the DSN of the specified image copy data set is defined with SHRLEVEL=REFERENCE and TYPE=FULL and is an image copy of the specified table. If the DSN is not defined with these properties, the session fails.</p> <p>If selected, PowerExchange reads the Image Copy Dataset regardless of the values of SHRLEVEL and TYPE and without verifying that the object ID in the image copy matches the object ID in the DB2 catalog.</p>
Filter Overrides	<p>Filters the source data that PowerExchange reads based on specified conditions. PWXPC adds filter conditions specified to the WHERE clause on the SELECT SQL statement and passes the SQL statement to PowerExchange for processing. You can use any filter condition syntax that PowerExchange supports for NRDB SQL. For more information, see the <i>PowerExchange Reference Manual</i>.</p> <p>For example, you can select records where a column called TYPE has a value of A or D by specifying the following filter condition:</p> <pre>TYPE='A' or TYPE='D'</pre>
SQL Query Override	<p>Overrides the SQL query sent to PowerExchange, including any filter overrides.</p> <p>Caution: For DB2 for z/OS data sources, PowerExchange automatically appends FOR FETCH ONLY to SQL SELECT statements. If you include FOR FETCH ONLY in the Sql Query Override attribute in the Properties area, the expression is included twice in the SELECT statement, and PowerExchange issues an error.</p>

6. Click **OK**.

RELATED TOPICS:

- [“Filtering Source Data with PWXPC” on page 205](#)

Extracting CDC Data in Change and Real-time Modes

To extract change capture data in change mode (CAPX), you select a change mode application connection. To extract data in real-time mode (CAPXRT), you select a real-time mode application connection.

Nonrelational source definitions imported in PowerCenter 8.x automatically have the appropriate Batch Reader selected for the source type. This reader selection cannot be changed.

RELATED TOPICS:

- [“CDC Application Connection Attributes” on page 113](#)
- [“Restart and Recovery Overview” on page 159](#)
- [“Commit Processing for CDC Sessions” on page 157](#)
- [“Understanding PWXPC Restart and Recovery” on page 160](#)
- [“Configuring the Restart Token File” on page 169](#)
- [“Configuring CDC Sessions for Restart Processing” on page 174](#)
- [“Enabling Recovery for a CDC Session” on page 173](#)

PowerExchange Extraction Maps

For CDC sessions, import source metadata from PowerExchange extraction maps to define source definitions that can be used in mappings.

Extraction maps are generated by the PowerExchange Navigator when you create a capture registration for a source. If necessary, you can modify an extraction map or create another extraction map based on an existing capture registration.

When you import an extraction map in PowerCenter Designer, select the **CDC Datamaps** option in the **Import from PowerExchange** dialog box. Designer then creates the source definition based on the extraction map. The **Metadata Extensions** tab for the source definition shows the schema name and map name imported from the extraction map. You do not need to specify this information in the session properties.

Alternatively, you can import source metadata from a relational database to define source definitions. However, you must then manually specify the extraction map name in the session **Properties** to use a CDC reader and an application CDC connection.

Note: For nonrelational sources, you must import an extraction map to use a CDC reader. You cannot select a CDC reader for sources that have a **Database type** value of `PWX_source_NRDB2`.

If you want to extract change data from a multi-record nonrelational source with extraction maps, you must create a PowerExchange capture registration for every table in the data map. For each registration, the PowerExchange Navigator creates a corresponding extraction map. You can then either import the data map as a multi-record nonrelational source (for batch usage) or import the extraction map for each table (for CDC usage).

Data Extraction from Multiple CDC Sources in a Session

If a mapping contains multiple sources from which you want to extract CDC data in change mode or real-time mode, all sources must use the same application connection.

Specify the application connection only on the first source in the mapping. All of the other connections can have a **Type** value of **None**.

Note: All of the sources must have the same source datatype. Sessions with mappings that include sources with multiple datatypes fail at run time.

Session Properties for Change and Real-time Mode Sessions

When you configure a session to extract data in Change or Real-time mode, PWXPC uses the extraction map to extract the source data. Source definitions for CDC sources can be created with extraction maps (CDC Datamaps in the Import dialog box) or by importing metadata from the relational database. The session properties will differ based upon how the source metadata was imported.

Configuring a Change or Real-time Session for Extraction Map Sources

Configure a Change or Real-time session for extraction map sources from the Task Developer.

1. In the Task Developer, double-click the session.

The **Edit Tasks** dialog box appears.

2. On the **Properties** tab, edit the following fields:

- In the **Commit Type** field, select **Source**.
- Clear the **Commit on End of File** property.
- To enable recovery for the session, set the **Recovery Strategy** property to **Resume from last checkpoint**. This setting is important for CDC sessions because it ensures that data and restart tokens are properly handled.

Note: With this recovery strategy, if you use a transformation for which the **Output is Deterministic** option is available, you must select this option. If you use a Lookup transformation, select the **Lookup Source is Static** or **Lookup Cache Persistent** option instead. If you do not select one of these options, you will not be able to validate the session in Workflow Manager.

- If you write change data to a relational target, ensure that the **Session Retry on Deadlock** property is *not* selected in the **Performance** section to prevent retries of target writes when a deadlock occurs. By default, this property is not selected. Also, in the relational connection definition for the target, ensure that the **Connection Retry Period** attribute is set to the default value of 0 to prevent database connection retries. Changing either of these settings from the default values requires enough buffers in the DTM buffer pool to hold the largest transaction that the session processes.

If you do not use the default settings for the **Session Retry on Deadlock** property and **Connection Retry Period** attribute, the CDC session might hang with the following message because not enough buffers are available:

```
WRT_8450 : The writer is waiting for buffer blocks. To improve session
performance and to prevent the session from hanging, increase the DTM buffer pool
size in the session properties.
```

In this case, set the **Commit Interval** session property and the **Maximum Rows Per commit** attribute on the source connection to the same value, for example, 1000. Although the **Commit Interval** property does not control when the session commits data to targets, it does affect the number of buffers in the DTM buffer pool that can be used to hold a transaction. Use the **Maximum Rows Per**

commit connection attribute to divide the rows in a large transaction into subsets for subpacket commit processing on the target. For more information about controlling commit processing, see [“Connection Attributes for CDC Commit Processing ” on page 122.](#)

3. On the **Mapping** tab, click the **Sources** view.
4. In the right pane under **Readers**, for relational sources, select the appropriate CDC Reader.
With extraction map sources, PWXPC automatically selects the reader based on the source type of the extraction map.
5. Under **Connections**, in the **Value** field, select **CDC Real Time** or **CDC Change** application connection type.
PWXPC displays the valid connections for the source type in the Application Connection Browser.
Optionally, you can open the application connection to override any connection values.
6. Under **Properties**, you can configure optional attributes.

The following table describes the optional attributes that you can configure:

Attribute Name	Description
Schema Name Override	Overrides the schema name in the PowerExchange extraction map.
Map Name Override	Overrides the PowerExchange extraction map name.
ADABAS Password	For the Adabas source type, an Adabas password for the source file. If the Adabas FDT for the source file is password-protected, enter the Adabas FDT password.
Database Id Override	For the Adabas source type, overrides the Adabas database ID in the PowerExchange data map.
File Id Override	For the Adabas source type, overrides the Adabas file ID in the PowerExchange data map.
Library/File Override	For the DB2i50S Real Time source type, overrides the library and file names in the extraction map. Specify the full library name and file name in the format: <i>library/file</i> Alternatively, specify an asterisk (*) wildcard for the library name to retrieve changes for all files of the same file name across multiple libraries. This attribute overrides the Library/File Override attribute on the application connection.
Source Schema Override	For the Oracle source type, overrides the source schema name.

Attribute Name	Description
Filter Overrides	<p>Filters the source data that PowerExchange reads based on specified conditions. PWXPC adds filter conditions specified to the WHERE clause on the SELECT SQL statement and passes the SQL statement to PowerExchange for processing. You can use any filter condition syntax that PowerExchange supports for NRDB SQL. For more information, see the <i>PowerExchange Reference Manual</i>.</p> <p>For example, you can select records where a column called TYPE has a value of A or D by specifying the following filter condition: TYPE='A' or TYPE='D'</p> <p>To select change records where columns ID and ACCOUNT have changed, you can use the DTL__CI columns by specifying the following filter condition: DTL__CI_ID='Y' and DTL__CI_ACCOUNT='Y'</p>
SQL Query Override	Overrides the SQL query sent to PowerExchange, including any Filter Overrides.

- Click **OK**.

Configuring a Change or Real-time Session for a Relational Source

Configure a Change or Real-time session for a relational source from the Task Developer.

- In the Task Developer, double-click the session.

The **Edit Tasks** dialog box appears.

- Click the **Properties** tab and change the following fields:

- In the **Commit Type** field, select **Source**.
- Clear the **Commit on End of File** option.
- To enable recovery for the session, set the **Recovery Strategy** property to **Resume from last checkpoint**. This setting is important for CDC sessions because it ensures that data and restart tokens are properly handled.

Note: With this recovery strategy, if you use a transformation for which the **Output is Deterministic** option is available, you must select this option. If you use a Lookup transformation, select the **Lookup Source is Static** or **Lookup Cache Persistent** option instead. If you do not select one of these options, you will not be able to validate the session in Workflow Manager.

- If you write change data to a relational target, ensure that the **Session Retry on Deadlock** property is *not* selected in the **Performance** section to prevent retries of target writes when a deadlock occurs. By default, this property is not selected. Also, in the relational connection definition for the target, ensure that the **Connection Retry Period** attribute is set to the default value of 0 to prevent database connection retries. Changing either of these settings from the default values requires enough buffers in the DTM buffer pool to hold the largest transaction that the session processes.

If you do not use the default settings for the **Session Retry on Deadlock** property and **Connection Retry Period** attribute, the CDC session might hang with the following message because not enough buffers are available:

```
WRT_8450 : The writer is waiting for buffer blocks. To improve session
performance and to prevent the session from hanging, increase the DTM buffer pool
size in the session properties.
```

In this case, set the **Commit Interval** session property and the **Maximum Rows Per commit** attribute on the source connection to the same value, for example, 1000. Although the **Commit Interval**

property does not control when the session commits data to targets, it does affect the number of buffers in the DTM buffer pool that can be used to hold a transaction. Use the **Maximum Rows Per commit** connection attribute to divide the rows in a large transaction into subsets for subpacket commit processing on the target. For more information about controlling commit processing, see [“Connection Attributes for CDC Commit Processing ” on page 122.](#)

3. Click the **Sources** view on the **Mapping** tab.

In the **Readers** field, select one of the following CDC readers based on the source type :

- For DB2 for z/OS, select either PowerExchange CDC Change or PowerExchange CDC Real-time for DB2zOS.
- For DB2 for i5/OS, select either PowerExchange CDC Change or PowerExchange CDC Real-time for DB2i5OS.
- For DB2 for Linux, UNIX, and Windows, select either PowerExchange CDC Change or PowerExchange Real-time for DB2LUW.
- For Oracle, select PowerExchange CDC Real-time for Oracle.
- For Microsoft SQL Server, select either PowerExchange CDC Change or PowerExchange Real-time for MSSQL.

4. In the **Connection Value** field, select an application connection.

Tip: For CDC Change and Real-time readers, select an application connection only for the first CDC source. For subsequent CDC sources, select a **Connection Type** of **None**. PowerExchange group source processing only uses the connection information from the first application connection. Subsequent application connection specifications are not required and might cause session failures.

5. Optionally, open the application connection to override any connection values.
6. In the **Properties** settings, configure the attributes that apply to the source type.

The following table describes required and optional attributes:

Attribute Name	Description
Extraction Map Name	Required. The PowerExchange extraction map name for the CDC source. You must specify the extraction map name for the relational source.
Library/File Override	Optional. For the DB2i5OS Real Time source type, overrides the library and file names in the extraction map. Specify the full library name and file name in the format: <i>library/file</i> Alternatively, specify an asterisk (*) wildcard for the library name to retrieve changes for all files of the same file name across multiple libraries. This attribute overrides the Library/File Override value on the application connection.
Source Schema Override	Optional. For the Oracle Change and Real Time source types, overrides the source schema name.

7. Click **OK**.

RELATED TOPICS:

- [“Connections” on page 58](#)
- [“Asterisk Wildcard in i5/OS Library or File Overrides” on page 142](#)
- [“Restart and Recovery” on page 159](#)

- [“Filtering Source Data with PWXPC” on page 205](#)

Asterisk Wildcard in i5/OS Library or File Overrides

For real-time extraction, if an i5/OS file exists in many libraries with the same name and layout, you can specify the asterisk (*) wildcard for the name in the **Library/File Override** attribute. PowerExchange can then read changes for the specified file name that are recorded in the same journal, across all of the libraries that contain the file.

With the asterisk wildcard, you configure only a single source in the mapping.

To enter the asterisk wildcard in the **Library/File Override** property, use the following syntax:

```
*/file_name
```

Consider the following rules and guidelines:

- You can use the asterisk wildcard in the **Library/File Override** property only for real-time extractions.
- DB2 for i5/OS files that are included in a mapping require PowerExchange capture registrations. By using the asterisk as a library override, you can map and register a single file for change data capture in the PowerExchange Navigator. CDC sessions can extract changes for all of the file instances that occur across all libraries.
- To extract changes for multiple file instances, the files must use the same journal and be defined with IMAGES(*BOTH).
- The specified file name must be a valid i5/OS file name. You cannot specify DB2 table names.
- If you use i5/OS-generated file names, make sure that you understand the journal contents. An i5/OS-generated file name is composed of only the first 5 characters of the DB2 table name followed by 5 digits. These file names might result in data extraction from unexpected files.
- Each time a real-time extraction session runs, PowerExchange checks the journal for files that match the file name. The files for which changes are extracted can vary from session to session if you add new libraries that contain the same file name and use the same journal.
- Restart tokens are only for sources that are included in a mapping. PWXPC indicates the presence of a **Library/File Override** value for a session by prefacing the extraction map name with the specified override.

The following example shows the restart tokens for a source that is processed by a session for which the **Library/File Override** attribute specifies **/file_name*:

```
<*/file_name>d3instance.regname_tablename=restart1
<*/file_name>d3instance.regname_tablename=restart2
```

You do not need to include the **Library/File Override** information in the restart token file when overriding a restart token. PWXPC includes the override information if it is found in the session properties.

Treating Source Rows as Updates

PowerExchange extraction maps include the DTL_CAPXACTION column, which indicates whether the change is an insert, update, or delete. When you import an extraction map in Designer, PWXPC includes this column in the source definition. The PowerCenter Integration Service uses the column to determine which action PowerCenter applies to the row when the row reaches the target.

You might need to configure a session to treat each source row as an update, regardless of the value of the DTL_CAPXACTION column. For example, suppose the change stream includes multiple row inserts that have the same value for a non-key field in the source table. If this field is the key field of the target table, an attempt to insert the rows into the target table results in a primary key constraint violation.

To configure a session to move each source row to the target as an update, complete the following steps:

1. In the Task Developer, double-click the session to edit it.
2. Click the **Properties** tab.
3. For the **Treat Source Rows As** attribute, select **Data driven**.
4. In the Mapping Designer, add an Update Strategy transformation to the mapping.
5. Open the Update Strategy transformation.
6. Click the **Properties** tab.
7. Click the button next to the **Update Strategy Expression** field.
8. Enter an update strategy expression to always mark rows as updates.
9. Validate the expression and click **OK**.
10. Click **OK**.
11. Connect the ports in the Update Strategy transformation to another transformation or a target instance.

When you run the session, PowerCenter moves each source row to the target as an update if possible. If a row with a matching primary key value is not present in the target system, PowerCenter inserts the row.

Note: For CDC data sources, if you select **Insert**, **Update**, or **Delete** for the **Treat Source Rows As** property, the Integration Service ignores the selection and uses the action that is indicated in the DTL_CAPXACTION column.

Loading Data to PowerExchange Targets

You can load data to relational targets as well as to nonrelational targets.

Configuring Sessions to Load Data to Relational Targets

To load data to a relational target, select a relational database connection. You can then configure properties for session as you would any other relational target.

Note: DB2 for i5/OS tables can also be accessed as flat files. If the table has been imported into PowerExchange as a sequential data map, you can use a PWX NRDB Batch application connection to write to it.

To configure sessions to load data to relational targets:

1. In the Task Developer, double-click the session to edit it.
2. Click the Targets view on the Mapping tab.
3. In the Writers setting, select Relational Writer to run sessions with relational targets.
4. In the Connections Value field, select a relational database connection from one of the following types:
 - For DB2 for i5/OS, select PWX DB2i5OS.
 - For DB2 for Linux, UNIX, and Windows, select PWX DB2LUW.
 - For DB2 for z/OS, select PWX DB2zOS.
 - For Microsoft SQL Server, select PWX MSSQLServer.
 - For Oracle, select PWX Oracle
5. Configure any other session properties.

Note: If you select the **Truncate Table Option** session property for a PWX DB2zOS target connection, by default, PowerExchange issues a TRUNCATE statement to truncate table contents before loading new data.

If you specify the DB2TRUNCASDEL=Y override option in the **PWX Override** connection attribute for a DB2 for z/OS target, PowerExchange issues a DELETE statement instead of a TRUNCATE statement. The DB2TRUNCASDEL override applies only to DB2zOS target connections.

For DB2i5OS target connections, PowerExchange always issues a DELETE statement. For DB2LUW target connections, PowerExchange always issues a TRUNCATE statement.

6. Click **OK**.

Configuring Sessions to Load Data to Nonrelational Targets

PWXPC can load data to nonrelational targets. Nonrelational target types include Adabas, IMS, sequential data sets, flat files, and VSAM files.

Based on the target type, PowerCenter selects the correct PowerExchange Writer. After you select a PWX NRDB Batch application connection, you can configure session properties for the target.

For information about configuring sessions for multiple-record nonrelational targets, see [“Multiple-Record Write Session Properties for Targets” on page 152](#).

1. In the Task Developer, double-click the session to edit it.
2. On the **Mapping** tab, expand the **Targets** node in the **Transformations** tree. Then click a nonrelational target.

PowerCenter sets the writer value based on the selected target type.

3. Under the **Connections**, in **Value** field, select a PWX NRDB Batch application connection.
4. Under **Properties**, set the session attributes for the target type, as needed.

The following table describes these attributes in alphabetical order:

Attribute Name	Description
ADABAS Password	For the ADABAS target type, the Adabas file password. If the ADABAS FDT for the target file is password protected, enter the ADABAS FDT password.
BLKSIZE	For the SEQ target type on z/OS, the z/OS data set block size. Default is 0, which means use the best possible block size. If you select VB for the RECFM value, the actual block size might be up to four bytes greater than the value you specify for BLKSIZE .
DATACLAS	For the SEQ target type on z/OS, the z/OS SMS data class name.
Delete SQL Override	For the ADABAS and VSAM target types, overrides the default Delete SQL that is sent to PowerExchange.

Attribute Name	Description
Disp	<p>For the SEQ target type on z/OS, the z/OS data set disposition.</p> <p>Valid values:</p> <ul style="list-style-type: none"> - OLD - SHR - NEW - MOD <p>Default is MOD if the data set exists, and NEW if it does not.</p>
File Name Override	<p>For the SEQ and VSAM target types, overrides the data set or file name in the PowerExchange data map. Enter the complete data set or file name.</p> <p>For i5/OS, use the following format: <i>library_name/file_name</i>.</p>
IMS AM Override	<p>For the IMS target type, overrides the IMS access method in the imported data map for the target with the other allowable access method. The session then uses the override access method at run time.</p> <ul style="list-style-type: none"> - If you imported a target data map that specifies the DL/1 BATCH access method, enter O to override it with the IMS ODBA access method. For ODBA access, you must also specify the IMS PSBNAME Override and IMS PCBNAME Override attributes. - If you imported a target data map that specifies the IMS ODBA access method, enter D to override it with the DL/1 BATCH access method, which provides DL/I or BMP access. You must also specify the IMS PCBNUMBER Override attribute. <p>Important: Before you run the session with an access method override, ensure that you complete the PowerExchange configuration tasks for the new access method. For example, if the override is DL/1 BATCH, you must configure LISTENER and NETPORT statements in the DBMOVER member and configure the netport JCL. If the override is IMS ODBA, you must perform other configuration tasks. For more information, see "IMS Data Maps" in the <i>PowerExchange Navigator User Guide</i>.</p>
IMS PCBNAME Override	<p>For the IMS target type, if you imported an IMS ODBA data map for the target and did not override the access method, this value overrides the PCB name from the data map. If you specified ODBA access as an override in the IMS AM Override attribute, you must enter this value. A PCB name is required for ODBA access.</p> <p>A PCB name is not used for DL/I or BMP access.</p>
IMS PCBNUMBER Override	<p>For the IMS target type, if you imported a DL/1 BATCH data map for the target and did not override the access method, this value overrides the PCB number from the data map. If you specified DL/I or BMP access as an override in the IMS AM Override attribute, you must enter this value. A PCB number is required for DL/I or BMP access.</p> <p>A PCB number is not used for IMS ODBA access.</p>
IMS PSBNAME Override	<p>If you imported an IMS ODBA data map for the target and did not override the access method, this value overrides the PSB name from the data map. If you specified ODBA access as an override in the IMS AM Override attribute, you must enter this value. A PSB name is required for ODBA access.</p> <p>If you use DL/I batch or BMP access and specify this override, you must also specify the PSB=%PSBNAME substitution variable in the netport JCL. The override value then replaces the substitution variable in the JCL.</p> <p>If you specify the PSB=%1 substitution variable instead of PSB=%PSBNAME in the netport JCL, the session uses the PSB name in the NETPORT statement, if specified. In this case, you need a separate NETPORT statement for each PSB. To avoid exceeding the limit of ten NETPORT statements, use this override with %PSBNAME substitution variable instead.</p>

Attribute Name	Description
IMS SSID Override	<p>For the IMS target type, if you imported an IMS ODBA data map for the target and did not override the access method, use this value to override the IMS subsystem ID (SSID). If you specified ODBA access as an override in the IMS AM Override attribute, you must enter this value. An SSID is required for ODBA access.</p> <p>If you use the IMS DL/1 BATCH access method and a BMP netport job, you can use this override with the %IMSID substitution variable in the netport JCL. This override replaces the substitution variable to specify the IMS SSID to use for the session. By using the substitution variable and override together, you can use the same netport JCL to access multiple IMS environments, such as development, testing, and production environments.</p> <p>Note: An IMS SSID is not required for DL/I batch access to IMS data or for access to an IMS unload file.</p>
Initialize Target	<p>For the VSAM target type, select this option to have PowerExchange allow both inserts and updates into empty VSAM data sets.</p> <p>If this option is not selected, PowerExchange only allows inserts into empty VSAM data sets.</p>
Insert Only	<p>For the ADABAS and VSAM target types, processes updates and deletes as inserts.</p> <p>Note: You must select this option when the target has no keys.</p>
Insert SQL Override	<p>For all nonrelational target types, overrides the default Insert SQL sent to PowerExchange.</p>
LRECL	<p>For the SEQ target type on z/OS, the data set logical record length.</p> <p>This value is ignored if Disp is not MOD or NEW.</p> <p>Default is 256.</p> <p>If you select VB for the RECFM value, specify the maximum number of data bytes in a logical record for LRECL. PowerExchange adds 4 to this value for the record descriptor word (RDW).</p>
Map Name Override	<p>For all nonrelational target types, overrides the target PowerExchange data map name.</p> <p>Note: PWXPC sends the file name that is specified for the source in the mapping unless this name is overridden in the File Name Override attribute.</p>
MGMTCLAS	<p>For the SEQ target type on z/OS, the SMS management class name.</p> <p>This value is ignored if Disp is not MOD or NEW.</p>
MODELDCB	<p>for the SEQ target type on z/OS, the Model DCB for non-SMS-managed GDG data sets.</p> <p>This value is ignored if Disp is not MOD or NEW.</p>
Post SQL	<p>For all nonrelational target types, one or more SQL statements that are executed after the session runs with the target database connection.</p>
Pre SQL	<p>For all nonrelational target types, one or more SQL statements that are executed before the session runs with the target database connection.</p> <p>Note: In certain cases, you must specify the Pre SQL run once per Connection attribute along with the Pre SQL attribute.</p>

Attribute Name	Description
Pre SQL run once per Connection	<p>For all nonrelational target types, runs the SQL that you specify in the Pre SQL attribute only once for a connection.</p> <p>Select this attribute in either of the following cases:</p> <ul style="list-style-type: none"> - In the Pre SQL attribute for a session that uses writer partitioning, you specify a SQL statement such as CREATEFILE that can run only once for the session. If you do not select Pre SQL run once per Connection, the session tries to run the statement once for each partition. - In the Pre SQL attribute for a session that performs a multiple-record write, you specify a CREATEFILE statement that creates a new generation of a GDG or creates an empty file. If you do not select Pre SQL run once per Connection, the session creates a generation or tries to create a new empty file for each record that the session writes.
Primary Space	<p>For the SEQ target type on z/OS, the primary space allocation, in the units specified in the Space attribute.</p> <p>This value is ignored if Disp is not MOD or NEW.</p> <p>Default is 1.</p>
RECFM	<p>For the SEQ target type on z/OS, the z/OS record format. Valid values are F, V, FU, FB, VU, VB, FBA, and VBA.</p> <p>This value is ignored if DISP is not MOD or NEW.</p>
Schema Name Override	<p>For all nonrelational target types, overrides the schema name in the target PowerExchange data map.</p> <p>Note: PWXPC sends the file name for the source in the mapping unless this name is overridden in File Name Override attribute.</p>
Secondary Space	<p>For the SEQ target type on z/OS, the secondary space allocation, in the units specified in the Space attribute.</p> <p>This value is ignored if Disp is not MOD or NEW.</p> <p>Default is 1.</p>
Space	<p>For the SEQ target type on z/OS, the type of units for expressing primary or secondary space for z/OS data sets. Valid values are:</p> <ul style="list-style-type: none"> - CYLINDER - TRACK <p>This value is ignored if Disp is not MOD or NEW.</p> <p>Default is TRACK.</p>
STORCLAS	<p>For the SEQ target type on z/OS, the SMS storage class name.</p> <p>This value is ignored if Disp is not MOD or NEW.</p>
Truncate target option	<p>For the VSAM target type, truncates, or deletes, table contents before loading new data.</p> <p>Note: VSAM data sets must be defined with the REUSE option for this truncate option to function correctly.</p>
UNIT	<p>For the SEQ target type on z/OS, the z/OS unit type.</p> <p>This value is ignored if Disp is not MOD or NEW.</p> <p>Default is SYSDA.</p>
Update SQL Override	<p>For the ADABAS and VSAM target type, overrides the default Update SQL that is sent to PowerExchange.</p>

Attribute Name	Description
Upsert	For the ADABAS and VSAM target type, processes failed inserts as updates and updates as inserts.
VOLSER	For the SEQ target type on z/OS, the volume serial number. This value is ignored if Disp is not MOD or NEW.

5. Click **OK**.

Block Size and Record Size in Variable-Blocked Data Sets

When you select **VB** for the **RECFM** attribute to create a variable-blocked data set, the block size and the record size of the data set vary slightly from the values that you specify for **BS** and **LRECL**.

When you select **VB** for the **RECFM** attribute, the value that you specify for **LRECL** represents the maximum number of data bytes in a record. To reserve space for the record descriptor word (RDW), PowerExchange adds four bytes to the **LRECL** value that you specify to determine the actual record length in the new data set.

Also, PowerExchange might add up to four bytes to the value that you specify for **BS**. In this way, PowerExchange ensures that the actual block size value is at least four bytes greater than the actual record size value. These four bytes are reserved for the block descriptor word (BDW).

For example, suppose you define the following attributes:

Attribute	Value
RECFM	VB
LRECL	27994
BLKSIZE	27998

The resulting data set has the following attributes:

```
Record format . . . : VB
Record length . . . : 27998
Block size . . . . : 28002
```

In this example, PowerExchange increases the record size from 27994 to 27998. Also, to ensure that the block size is at least four bytes greater than the record size, PowerExchange increases the block size from 27998 to 28002.

Depending on the track size, the resulting block size might be greater than half a track, resulting in wasted space. To avoid this problem, ensure that the value that you specify for **LRECL** represents the number of data bytes in the record and does not include an additional four bytes for the RDW.

Considerations for Sessions that Perform Multiple-Record Writes

When you work with sessions that perform multiple-record writes, special considerations apply in the following areas:

- Sequencing and queuing cache for multiple-record writes
- Row statistics for multiple-record writes
- Multiple-record write session properties for sources
- Multiple-record write session properties for targets

RELATED TOPICS:

- [“Multiple-Record Writes to Nonrelational Targets” on page 50](#)
- [“Connection Attributes for Multiple-Record Writes” on page 126](#)

Sequencing and Queuing Cache for Multiple-Record Writes

For multiple-record write operations, PowerExchange maintains a cache, called the *sequencing and queuing cache*, on the PowerCenter Integration Service machine. PowerExchange uses this cache to queue data from nonrelational sources with multiple record types in the correct sequence before sending it to the target.

A PowerCenter workflow processes data from different source record types in different pipelines. The source records can represent IMS segments or sequential or VSAM records. The PowerExchange Writer receives data for a given record type in order, but data for different record types might arrive out of order.

Data for any given record type arrives and is written to the cache the same sequence as on the source. The relationship of data for any record type to other records is maintained by the following items:

- Primary key (PK). Contains a sequence number that indicates the physical sequence of the record in the source.
- Foreign Key (FK), the sequence number of the data record of which the current data is a child
- segment sequence number, which corresponds to the segment or record to which the data belongs

You can use the CSQ_MEMSIZE parameter of the PWX Override connection parameter to limit the memory that the cache uses. When this limit is reached, data is spilled to files on disk.

Separate threads write to and read from the cache.

The *producer thread* writes data records to the cache. The producer maintains one or more queues for each record.

The *consumer thread* reads data from the cache, uses the sequencing information to reassemble the data in the correct order, and writes the data to the target data set. The consumer thread reads records from the cache in primary key sequence. Reading from the cache correctly reflects the original order in which the data was read from the source.

If the consumer thread encounters any orphan records, it skips them and reports the number of records skipped. In this way, the consumer thread detects situations where data has been orphaned by a parent that was filtered out or dropped through the mapping logic.

PowerExchange reports the number of records that it writes to and reads from the cache in the following messages:

```
PWX-35805 Records written to cache: total_record_count
PWX-35806 Records read from cache: total_record_count
```

If PowerExchange skips any records, it reports the number of records that were cached, written, and skipped in the following message:

```
PWX-35807 Records cached = cached, records written = written, records skipped = skipped
```

PowerCenter also provides messages that involve the cache. For more information, see [“Row Statistics for Multiple-Record Writes” on page 151](#).

Notes:

- Because of its asynchronous nature, the PowerExchange writer might still be writing data after PowerCenter has processed its last commit.
- If a VSAM or sequential data set includes one or more header records, PowerExchange might write the entire file into the cache and write to the target only when all records have been processed.
- Because data for different record types are processed by different PowerCenter pipelines, the amount of data written into the cache largely depends on the arrival of data for each record type in PowerExchange. For example, a record type that does not have any data at all might force the entire file to be written into the cache before being processed.

Cache Spill Files

When the sequencing and queuing cache reaches its maximum memory size, it spills data to disk. The CSQ_MEMSIZE parameter of the PWX Override connection parameter defines the maximum memory size, in megabytes. Default is 16.

The caching module creates the spill files in the directory that is specified by one of the following environment variables:

- TMP environment variable on Windows
- TMPDIR environment variable on Linux and UNIX

If the environment variable is not set, the caching module writes the spill files to the current directory.

In some cases, the entire contents of the target file is first written to spill files on disk. To help you identify these files and troubleshoot potential problems, such as running out of disk space in the temporary directory, PowerExchange writes a series of messages to the message log.

PowerExchange writes some or all of the following messages when it spills data to disk:

- **PWX-35860.** Reports the following information:
 - Spill file directory, based on the value of the TMP or TMPDIR environment variable
 - Name of the environment variable, or "CWD" if no environment variable is specified
 - Amount of memory allocated to the cache, based on the value of CSQ_MEMSIZE
- **PWX-35861 to PWX-35864.** Report the allocation number and name of each spill file that is allocated, deallocated, assigned to a new transaction, or retained, respectively. The allocation number represents how many spill files have been allocated during the multiple-record write session.

To prevent excessive file deallocation and allocation activity, by default, three spill files are retained and are assigned to new transactions, as needed, during the session.
- **PWX-35853.** When a problem occurs in the caching module append() routine, reports the returned error code and number. An error can occur in this module if the cache requires an enormous amount of disk space or a permission problem occurs that prevents writing to the temporary directory.
- **PWX-35851, PWX-35854.** When an error occurs that PWX-35853 does not cover, PWX-35851 reports the module name and error code and number. Or if the queue name is relevant, PWX-35854 reports the module name, error code and number, and queue name.

Row Statistics for Multiple-Record Writes

The following statistics for multiple-record writes are reported in session logs and the Workflow Monitor:

- Requested row count. The number of rows that PWXPC has passed to the sequencing and queuing cache.
- Applied and affected row count. The number of rows that have been successfully read from the sequencing and queuing cache.
- Rejected row count. The number of rows that were rejected as orphans. Orphans are records for which no record exists in the cache with a primary key that matches the foreign key of the orphan.

Because multiple-record writes require the **Confirm Write Off** target connection setting, these writes are asynchronous with respect to PowerCenter. Depending on input data to the writer, row statistics reported in the session log and Workflow Monitor might not always reflect the number of rows that have actually been written to the target.

Multiple-Record Write Session Properties for Sources

For PowerCenter bulk data movement sessions that perform multiple-record writes, you can define the following session properties for the source:

Flush After N Blocks

Optional. By default PWXPC flushes blocks of data only when buffers are completely full or at end-of-file. Flushing might not occur frequently if some of the record types do not have as much data as others. A delayed flush can cause one of the seldom-occurring record types to have no data to be written to the target for a long time, thereby blocking flushing on the writer side.

To ensure that buffers for all records are flushed at a regular interval, define this session property. This property specifies the maximum number of blocks that can be flushed across all record types without a block for any one record type being flushed. A value of zero disables this feature. Flushing then occurs only when blocks are filled.

Valid values are -1 to 100000.

The default value of -1 works in the following manner:

- For multiple-record sources that do not use sequencing, the default value of Flush After N blocks = 0. That is, the feature is disabled by default.
- For multiple-record sources that use sequence fields, the default value of Flush After N blocks = $7 \times (\text{number of record types in the source})$.

Example

Assume a multiple-record source has four record types: A, B, C, and D. If **Flush After N Blocks** is not defined, the following blocks are flushed when their buffers are full:

C, C, C, B, C, B, C, C, C, C, C, B, A, D flush (all)

If **Flush After N Blocks** is set to 4, the following flushes occur:

C, C, C, B + ForcedFlush(A, D), C, B, C, C + ForcedFlush(A, D), C, C + ForcedFlush (B), C, A + ForcedFlush (D), Flush (all)

The terminology ForcedFlush(X) indicates that the flush is caused by the **Flush After N Blocks** setting and not by a block becoming full.

IMS Unload File Name

An IMS unload file name. Specify this attribute to read source data from an IMS unload file instead of from the IMS database. Required when the **Use Sequence Fields** option is selected for an IMS source definition.

File Name Override

Source file name. Optionally define this attribute for sequential or VSAM sources. This source file name takes precedence over the file name in the source definition or data map.

Multiple-Record Write Session Properties for Targets

When you define a PowerCenter session that performs a multiple-record write, you must specify the name of the target file in the target session properties. Optionally, enter the other session properties for the target.

File Name Override

Target sequential or VSAM file name. This file name overrides the file name in the target definition and data map.

IMS Unload File Name

Target IMS unload file name. Specify this attribute to write source data to an IMS unload file instead of an IMS database. Required when the **Use Sequence Fields** option is selected for an IMS target definition.

Insert SQL override

Overrides the SQL query for inserts that is sent to PowerExchange.

PWXPC replaces the default SQL query with the SQL statement that you enter. PWXPC passes this SQL statement to PowerExchange for processing. You can enter any SQL statement that PowerExchange supports for NRDB SQL.

For a multiple-record target with use sequence fields, the SQL query override must include the generated sequence fields in the query.

Use the following syntax:

```
group_name1=insert_sql_override1;    group_name2=insert_sql_override2;...
```

Map Name Override

Overrides the name of the PowerExchange data map for the target.

PWXPC sends the map name from the source definition unless this value overrides it.

Pre SQL

One or more SQL statements that PWXPC executes before the session runs with the target database connection.

If you specify a CREATEFILE command that creates a new generation of a GDG in the Pre SQL property, also select the **Pre SQL once per Connection** property. Otherwise, the session creates a generation for each record that the session writes.

Pre SQL run once per Connection

Runs the SQL that you specify in the **Pre SQL** property only once for a connection.

Select this property if the **Pre SQL** property includes a CREATEFILE statement that creates a new generation of a GDG or creates an empty file. Otherwise, the session creates a generation or tries to create a new empty file for each record that the session writes.

Post SQL

One or more SQL statements that PWXPC executes after the session runs with the target database connection.

Schema Name Override

Overrides the schema name of the PowerExchange data map for the target.

PWXPC sends the schema name from the source definition unless this value overrides it.

SEQ Properties on z/OS

For SEQ targets on z/OS, you can optionally define the following data set allocation properties to have the session allocate the target data set when the session runs:

- Disp
- Space
- Primary Space
- Secondary Space
- LRECL
- BLKSIZE
- RECFM
- UNIT
- VOLSER
- MGMTCLAS
- DATACLAS
- STORCLAS
- MODELDCB

For more information about session properties for targets, see [“Configuring Sessions to Load Data to Nonrelational Targets” on page 144](#).

Constraint-Based Loading for Relational Targets

In the Workflow Manager, you can specify constraint-based loading for a session. When you select this option, the PowerCenter Integration Service orders the target load on a row-by-row basis. For every row generated by an active source, the PowerCenter Integration Service loads the corresponding transformed row first to the primary key table and then to any foreign key tables.

Configuring Constraint-Based Loading for Relational Targets

In the Workflow Manager, you can specify constraint-based loading for a session. When you select this option, the PowerCenter Integration Service orders the target load on a row-by-row basis. For every row generated by an active source, the PowerCenter Integration Service loads the corresponding transformed row first to the primary key table, then to any foreign key tables.

The **Constraint based load ordering** attribute applies only to insert operations. Change data normally contains a mixture of insert, update, and delete operations. PowerCenter provides a custom property to enable complete constraint-based loading.

When you enable complete constraint-based loading, change data is loaded to targets in the same Transaction Control Unit (TCU) by using the row ID assigned to the data by the CDC Reader. As a result, data is applied to the targets in the same order in which it was applied to the sources. The following message will be issued in the session log to indicate that this support is enabled:

```
WRT_8417 Complete Constraint-Based Load Ordering is enabled.
```

To enable complete constraint-based loading, specify **FullCBLOSupport=Yes** in the **Custom Properties** attribute on the **Config Object** tab. This property can also be set in the PowerCenter Integration Service, which makes it applicable to all workflows and sessions that use that the PowerCenter Integration Service.

If you use complete constraint-based loading, your mapping must not contain active transformations which change the row ID generated by the CDC Reader. The following transformations change the row Id value:

- Aggregator
- Custom, configured as an active transformation
- Joiner
- Normalizer
- Rank
- Sorter

Variables for Session and Connection Properties

You can use variables to assign values to properties in the Designer and Workflow Manager and to override some service and service process properties.

You can also use variables to override PWXPC session properties. You can also use variables to override source or target relational or application connection attributes other than PWXPC connection attributes.

PWXPC variables must begin with \$PWX.

The following table lists the PWXPC session source and target properties for which you can specify variables:

Session Property	Source or Target Type
DATACLAS	MVS sequential file sources and targets
DB2 Sub System Id	DB2 image copy and DB2 data map sources
DB2 Table name	DB2 data map sources
Delete SQL override	Nonrelational targets
Extraction Map Name	Change data capture sources
File Name Override	VSAM and sequential sources and targets
Filter Overrides	All sources
Image Copy Dataset	DB2 image copy sources
IMS Unload File Name	IMS sources

Session Property	Source or Target Type
Insert SQL override	Nonrelational targets
Map Name Override	All sources and nonrelational targets
MGMTCLAS	MVS sequential file sources and targets
MODELDCB	MVS sequential file sources and targets
Post SQL	Nonrelational targets
Pre SQL	Nonrelational targets
Schema Name Override	All
Source Schema Override	Relational change data capture sources and DB2 image copy
Sql Query Override	All sources
STORCLAS	MVS sequential file sources and targets
UNIT	MVS sequential file sources and targets
Unload File Name	Adabas and DB2 unload file sources
Update SQL override	Nonrelational targets

Workflows that Connect to a PowerExchange Listener Service

When you run a workflow, the PowerCenter Integration Service can connect to the PowerExchange Listener through the Listener Service. The Listener Service is an application service in the Informatica domain that manages a PowerExchange Listener. The Listener Service can define backup nodes to run a PowerExchange Listener process if the PowerExchange Listener on the primary node becomes unavailable.

The PowerCenter Integration Service connects to the PowerExchange Listener based on the way you configure the NODE statement in the DBMOVER configuration file:

- If the NODE statement includes the SERVICE_NAME parameter, the PowerCenter Integration Service connects to the Listener through the Listener Service.
- If the NODE statement does not include the SERVICE_NAME parameter, the PowerCenter Integration Service connects directly to the Listener. It does not connect through the Listener Service.

For more information about the Listener Service, see the *Informatica Administrator Guide*.

Configuring PowerExchange Services for Workflows

Before you run a workflow that uses PowerExchange services, you must configure the services.

To configure PowerExchange Services for workflows:

1. Install PowerExchange on the PowerCenter Integration Service machine and on each node where you plan to run the Listener Service and Logger Service.
2. Customize the DBMOVER file on each node where you plan to run PowerExchange Services, as follows:
 - Configure the LISTENER statement to identify the PowerExchange Listener.
 - For CDC sessions, configure CAPI_CONNECTION statements to control extraction processing. Include the same statements in each DBMOVER file.
 - If the PowerExchange Listener will connect to the PowerExchange Logger for Linux, UNIX, and Windows, customize the settings required for the PowerExchange Logger.
3. Customize the DBMOVER file on each node that runs the PowerCenter Integration Service. Configure the SERVICE_NAME parameter of the NODE statement to identify the Listener Service.
4. If the PowerExchange Listener for the data source will connect to a PowerExchange Logger, customize the pwxcl.cfg file on each node where you plan to run the Logger Service. Configure each pwxcl.cfg file to point to the same PowerExchange Logger files, and verify that the files are accessible to each node.
5. Create a PowerExchange Listener Service on the primary and backup nodes for the service.
6. If the PowerExchange Listener for the data source will connect to a PowerExchange Logger, create a PowerExchange Logger Service on the primary and backup nodes for the service.
7. Start the services in the Administrator tool.

Pipeline Partitioning

You can use pipeline partitioning at reader and writer partition points in bulk data movement sessions. Reader partitioning supports several partition types for data sources. Writer partitioning supports pass-through partitioning for VSAM and sequential file targets. For more information, see the *PowerExchange Bulk Data Movement Guide* and the *PowerCenter Advanced Workflow Guide*.

You can also use pipeline partitioning for targets in CDC sessions. However, you cannot use partitioning for sources in CDC sessions.

Commit Processing for Bulk Data Movement Sessions

During a bulk data movement session, by default PowerExchange commits data to the target each time the commit interval that is defined in the session properties is reached.

You might need to change how frequently PowerExchange commits data to the target. If commits are issued too infrequently, bulk data movement sessions with a nonrelational target might fail with an out-of-memory error.

To prevent or correct this problem, perform one of the following actions:

- Reduce the Commit Interval session property. The default is 10,000 rows.
- Tune the database based on the error message that you receive.
- If necessary, you can define ENABLE_AUTOCOMMIT=Y in the DBMOVER configuration file on the client machine. Defining ENABLE_AUTOCOMMIT=Y instructs PowerExchange to commit data to the target each time the data buffer becomes full. 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.

Caution: Defining ENABLE_AUTOCOMMIT=Y alters the standard commit and rollback behavior for PowerCenter sessions. Avoid using this statement if possible.

Commit Processing for CDC Sessions

The PowerCenter Integration Service, in conjunction with PWXPC, commits data to the target based on commit connection attributes and the commit type.

By default, the **Commit Type** property on the session Properties tab specifies **Target**. For CDC sessions, the Integration Service always uses source-based commit processing. When you run a CDC session that specifies target-based commit processing, the PowerCenter Integration Service automatically changes the commit type to source-based and writes message WRT_8226 in the session log.

For CDC sessions, PWXPC ignores the **Commit Interval** attribute. To control commit processing, configure attributes on the PWX CDC Change and Real Time application connections.

RELATED TOPICS:

- [“Connection Attributes for CDC Commit Processing ” on page 122](#)

Stored Procedure Calls in Pre- and Post-Session SQL Commands

You can specify pre- and post-session SQL in the Source Qualifier transformation and the target instance when you create a mapping. When you create a Session task in the Workflow Manager you can override the SQL commands on the Mapping tab.

For the following databases, if you include a call to a stored procedure in the pre- or post-session SQL, PowerExchange issues error message PWX-00183:

- DB2 for Linux, UNIX, and Windows
- Microsoft SQL Server
- Oracle

For best performance, if a Source Qualifier transformation for a PowerExchange DB2 for i5/OS source in a PowerCenter session calls a stored procedure on i5/OS to return data, specify the following options on the CREATE PROCEDURE statement when you create the stored procedure:

```
SET OPTION DATFMT = *ISO ,TIMFMT = *ISO
```

These options ensures that the format of date and time column types are returned in the same format as if the data was being returned through the DB2 database.

Preserving Low Values in Source Character Fields

If a data source includes character fields with hexadecimal '0' values, called *low values*, you can configure PowerExchange and PowerCenter to preserve these values so that a PowerCenter session can write the low values to a VSAM target on z/OS or to a sequential file target on Linux, UNIX, Windows, or z/OS.

To preserve low values, perform the following configuration tasks:

- Configure a PowerCenter session that uses the PowerExchange Client for PowerCenter (PWXPC) instead of the PowerExchange ODBC interface.
- If you want to write low values from a nonrelational data source on z/OS to a nonrelational target on z/OS without any translation of the values, complete the following tasks in PowerCenter:
 - On the **Config Object** tab for the session, enter **PreserveLowValues=Yes** in the **Custom Properties** field.
 - In the source and target PWX NRDB Batch application connections that are included in the PowerCenter workflow, ensure that the **Convert character data to string** option is *not* selected.
 - In the DBMOVER configuration files on the source system and the Integration Service system, set the LOWVALUES statement to Y. For more information, see the *PowerExchange Reference Manual*.

If you do not perform these configuration tasks, PowerExchange does not preserve low values. It interprets a hexadecimal '0' value as the end of a column string and pads the remaining length of the string with spaces. The portion of the column string after the hexadecimal '0' value is not written to the target.

CHAPTER 6

Restart and Recovery

This chapter includes the following topics:

- [Restart and Recovery Overview, 159](#)
- [Understanding PWXPC Restart and Recovery, 160](#)
- [Creating Recovery Tables, 167](#)
- [Configuring the Restart Token File, 169](#)
- [Using PWXPC Restart and Recovery, 172](#)

Restart and Recovery Overview

PWXPC supports restart and recovery processing. Learn how to configure your CDC sessions to use this processing.

Each source in a CDC session has unique restart information, referred to as restart tokens. PWXPC manages the CDC restart information. The PowerCenter Integration Service provides recovery for the target files and tables in CDC sessions.

To extract change data from the change stream, PWXPC provides restart information for CDC sources to PowerExchange. PowerExchange reads the change stream on the source platform and provides complete units of work (UOWs) to PWXPC. A UOW is a collection of changes within a single commit scope, which are made by a transaction on the source system. Based on the commit interval information that is specified on the CDC session connection, PWXPC periodically flushes complete UOWs to the PowerCenter Integration Service.

Target recovery and restart information is stored as the target tables and files are updated by the PowerCenter Integration Service. The PowerCenter Integration Service and PWXPC use this information to recover and restart stopped or failed sessions from the point of interruption.

RELATED TOPICS:

- [“CDC Application Connection Attributes” on page 113](#)
- [“Commit Processing for CDC Sessions” on page 157](#)
- [“Understanding PWXPC Restart and Recovery” on page 160](#)
- [“Configuring the Restart Token File” on page 169](#)
- [“Configuring CDC Sessions for Restart Processing” on page 174](#)
- [“Using PWXPC Restart and Recovery” on page 172](#)
- [“Enabling Recovery for a CDC Session” on page 173](#)

- [“Extracting CDC Data in Change and Real-time Modes” on page 137](#)

Understanding PWXPC Restart and Recovery

PowerCenter and PWXPC provide restart and recovery functionality.

Session Recovery

To use PowerCenter and PWXPC restart and recovery functionality for a session, select the recovery strategy of **Resume from the last checkpoint** in the session properties. This recovery strategy ensures that both the target data and CDC restart information can be recovered in the event of a session failure.

Warning: Do not enable recovery processing if any of the targets in the CDC session use the File Writer to write CDC data to flat files. The restart tokens for all targets in the session, including relational targets, will be compromised if a flat file target is in the same session. In this case, data loss or duplication might occur.

When you enable a resume-from-last-checkpoint recovery strategy, the PowerCenter Integration Service provides recovery for the target tables and files and PWXPC provides recovery for the CDC restart information. PWXPC issues the following message to indicate that recovery is in effect:

```
PWXPC_12094 [INFO] [CDCRestart] Advanced GMD recovery in affect. Recovery is automatic
```

The PowerCenter Integration Service stores the session state of operation in the shared location \$PMStorageDir. The PowerCenter Integration Service saves relational target recovery information in the target database.

CDC restart information consists of restart tokens that originate from PowerExchange on the CDC source platform. PWXPC stores CDC restart information in the following locations depending on the target type:

- For nonrelational targets, PWXPC stores the CDC restart information in state files in the shared location \$PMStorageDir on the Integration Service system.
- For relational targets, PWXPC stores the CDC restart information in state tables in the target database.

When the PowerCenter Integration Service performs recovery, it restores the state of operation to recover the session from the point of interruption. The Integration Service uses the target recovery information to determine how to recover the target tables. PWXPC and PowerExchange use the CDC restart information to determine the correct point in the change stream from which to restart the extraction.

Recovery Tables

For relational targets, the PowerCenter Integration Service creates the following recovery tables in the target database:

- **PM_RECOVERY.** This table contains target load information for the session run. The PowerCenter Integration Service removes the information from this table after each successful session and initializes the information at the beginning of subsequent sessions.
- **PM_TGT_RUN_ID.** This table contains information the Integration Service uses to identify each target on the database. The information remains in the table between session runs. If you manually create this table, you must create a row and enter a value other than zero for LAST_TGT_RUN_ID to ensure that the session recovers successfully.

- **PM_REC_STATE.** This table contains restart information for CDC sessions. The restart information recorded in the table contains the application name and restart tokens for the session. The restart information remains in the table permanently. The PowerCenter Integration Service updates it with each commit to the target tables.

If you edit or drop the recovery tables before you recover a session, the PowerCenter Integration Service cannot recover the session. If you disable recovery, the PowerCenter Integration Service does not remove the recovery tables from the target database. You must manually remove the recovery tables.

If you want the PowerCenter Integration Service to create the recovery tables, grant table creation privilege to the database user name for the target database connection.

If you do not want the PowerCenter Integration Service to create the recovery tables, create the recovery tables manually.

Tip: If you use PowerExchange relational target connections, you must manually create these tables so you can assign the desired database attributes.

RELATED TOPICS:

- [“Recovery Table Creation with PowerExchange Targets” on page 167](#)
- [“Creating the Recovery Tables Manually” on page 168](#)

Recovery State Table

The recovery state table contains the state and CDC restart information for a CDC session. The table resides in the same target database as target tables. The PowerCenter Integration Service creates the state table automatically if it does not exist.

The PowerCenter Integration Service creates an entry in the state table for each CDC session. These entries may comprise more than one row. CDC sessions with heterogeneous relational target tables have entries in the state table in each unique target database. For example, a CDC session which targets Oracle and SQL Server tables has an entry in the state table in the target Oracle database instance and in the target SQL Server database.

Each session entry in the state table contains a number of repository identifiers as well as execution state data like the checkpoint number and the CDC restart information. The following columns contain PWXPC-specific restart information:

- The APPL_ID column contains the application name specified in the source application connection, with the task instance ID appended.
- The STATE_DATA column, which contains the restart tokens for the session, is a variable 1,024-byte binary column. If the number of restart tokens for a session causes the data to exceed 1,024 in length, additional rows are added to accommodate the remainder of the restart information. The SEQ_NUM field is increased by one, starting from zero, for each additional row added for a session entry.

The majority of the columns in the table are task and workflow repository attributes. These repository attributes remain static unless the task or workflow is altered. The following examples are actions that alter these repository attributes:

- Adding or removing sources or targets from the mapping used by the session
- Moving the workflow or session to a different folder
- Moving the session to a different workflow

The PowerCenter Integration Service uses the APPL_ID column to retrieve CDC restart information. Since the APPL_ID column contains the application name and the task instance ID for the session, adding and removing sources or targets from the session affects restart processing.

During session initialization, the PowerCenter Integration Service reads the state table looking for an entry that matches APPL_ID information for the CDC session. If a match is found, the PowerCenter Integration Service uses that entry for target recovery processing. PWXPC uses the CDC restart information stored in the STATE_DATA column to perform restart and recovery processing.

RELATED TOPICS:

- [“Creating Recovery Tables” on page 167](#)
- [“Changing CDC Sessions and Restarting from the Point of Interruption” on page 178](#)

Recovery State File

The recovery state file is similar to the state table. PWXPC uses the state file to store the CDC restart information for nonrelational target files. Nonrelational target files include MQ Series message queues, PowerExchange nonrelational targets, and other PowerCenter nonrelational targets. The PowerCenter Integration Service automatically creates the recovery state file in the shared location, \$PMStorageDir, if it does not exist. The file name for the recovery state file is prefixed with:

```
pm_rec_state_application_name
```

The PowerCenter Integration Service uses the application name from the source CDC connection for the *application name* value in the state file name prefix. The PowerCenter Integration Service includes the complete file name in message CMN_65003. The remainder of the fields in the file name are task and workflow repository attributes. These repository attributes remain static unless the task or workflow is altered. The following examples are actions that alter these repository attributes:

- Adding or removing sources or targets from the mapping used by the session
- Moving the workflow or session to a different folder
- Moving the session to a different workflow

RELATED TOPICS:

- [“Changing CDC Sessions and Restarting from the Point of Interruption” on page 178](#)

Restart Token File

PWXPC stores the restart tokens in the state table in the target database for relational targets and in the state file on the PowerCenter Integration Service platform for nonrelational targets. The restart token file provides:

- Initial restart tokens for new CDC sessions
- Overrides for the restart tokens in the state table or file for existing CDC sessions

PWXPC uses the restart token file in the folder specified in the **RestartToken File Folder** attribute of the source CDC connection. PWXPC automatically creates this folder, if it does not exist, when the attribute contains the default value of \$PMRootDir/Restart. PWXPC does not automatically create any other restart token folder name.

During session initialization, PWXPC:

- Uses the name specified in the **RestartToken File Name** attribute to create an empty restart token file, if one does not already exist.

Creates a merged view of the restart tokens by reconciling the restart tokens specified in the restart token file with those in state tables and the state file for all relational and non-relations targets, respectively.

- Places the results of the restart token reconciliation process into an initialization file in the restart token file directory and empties out the restart token file.

Emptying the restart token file ensures that it does not override the state table or state file restart tokens with the same restart information the next time the session is run.

During normal termination, PWXPC writes the ending restart tokens into a termination file in the restart token file directory. The restart token files containing the initialization and termination restart tokens have the following names:

```
<restart_token_file_name>yyyymmddhhmmss_init
<restart_token_file_name>yyyymmddhhmmss_term
```

Where:

- *restart_token_file_name* is the restart token file name from the CDC connection
- *yyyymmddhhmmss* is the initialization file creation time stamp
- *init* or *term* is for initialization and termination files, respectively

For example, a CDC source application connection specifies a restart token file name of *my.app.txt*, which does not exist. PWXPC creates the following files on the PowerCenter Integration Service platform in the restart token file folder specified in the connection:

- *my.app.txt*
- *my.app.txtyyyymmddhhmmss_init*
- *my.app.txtyyyymmddhhmmss_term*

The restart token file, *my.app.txt*, is empty. The timestamps on both the initialization and termination files are the same to indicate that they are related to the same run. The termination file may not exist or may be empty if the session fails.

If you use the default value of zero for the connection attribute **Number of Runs to Keep RestartToken File**, PWXPC keeps only one copy of the paired initialization and termination files. Otherwise, PWXPC uses the value specified in that attribute to determine the number of backup copies of these paired files to keep. During termination, PWXPC removes any additional pairs of the backup files beyond the **Number of Runs to Keep RestartToken File** value.

RELATED TOPICS:

- [“How Restart Points Are Determined” on page 163](#)

How Restart Points Are Determined

The restart tokens PWXPC uses depends on how you start the CDC session and whether you provide any override restart tokens in the restart token file.

Based on the start type, PWXPC determines the restart point in one of the following ways:

- **Cold start.** When you cold start a CDC session, PWXPC reads only the restart token file to acquire restart tokens for all sources and does not attempt to recover the session. The session continues to run until stopped or interrupted.
- **Warm start.** When you warm start a CDC session, PWXPC reconciles the restart tokens provided in the restart token file, if any, with the restart tokens that exist in the state file or state tables. If necessary, PWXPC performs recovery processing. The session continues to run until stopped or interrupted.
- **Recover operation.** When you recover a CDC session, PWXPC reads the restart tokens from the state file and state tables and writes the tokens to the restart token file. If necessary, PWXPC performs recovery processing. After PWXPC finishes updating the restart token file and doing any necessary recovery, the session ends.

You can specify restart token overrides in the restart token file.

Each CDC source in the CDC session has its own unique restart point. Do not create and populate the restart token file with restart points for sources prior to running a CDC session for the first time. When restart tokens do not exist in the restart token file and no entry exists for the session in the state file or state tables, PWXPC passes null restart tokens to PowerExchange for all sources in the session.

RELATED TOPICS:

- [“Starting CDC Sessions” on page 176](#)

Cold Start Restart Points

You can cold start workflows and sessions with the cold start command from Workflow Manager, Workflow Monitor, or *pmcmd* program.

For a cold start, PWXPC disregards the state file and state tables and uses the restart token file as follows:

- If the restart token file is empty or does not exist, PWXPC assigns null restart tokens to all sources in the session.
- If the restart token file contains explicit override statements, PWXPC assigns the restart tokens in the explicit override statements to the specified sources. For all remaining sources, PWXPC assigns the oldest restart point based on the available restart tokens.
- If the restart token file contains a special override statement only, PWXPC assigns the restart tokens in the special override statement file to all sources.
- If the restart token file contains a special override statement and explicit override statements, PWXPC assigns the restart tokens in the explicit override statements to the appropriate sources. For all remaining sources, PWXPC assigns the restart tokens in the special override statement.

Warm Start Restart Points

You can warm start workflows and sessions with the start or restart command from Workflow Manager, Workflow Monitor, or *pmcmd* program.

PWXPC merges the restart tokens provided in the restart token file with any restart information that exists in the state file (nonrelational targets) or the state tables (relational targets), as follows:

- If the restart token file is empty or does not exist, the restart tokens are determined as follows:
 - If a state file or state table entry does not exist for the session, PWXPC assigns null restart tokens to all sources in the session.
 - If a state file or state table entry exists for some but not all sources in the session, PWXPC assigns the restart tokens in the state file or state tables to the specified sources. For all remaining sources that do not have specified restart tokens, PWXPC assigns the oldest restart point based on the available restart tokens.
 - If a state file or state table entry exists for all sources in the session, PWXPC uses the restart tokens from the state file or state tables.
- If the restart token file contains explicit override statements, the restart tokens are determined as follows:
 - If no state file or state table entry exists for the session, PWXPC assigns the restart tokens in the explicit override statements to the appropriate sources. For all remaining sources that do not have specified restart tokens, PWXPC assigns the oldest restart point based on the available restart tokens in the restart token file.
 - If a state file or state table entry exists for some but not all sources in the session, PWXPC assigns the restart tokens in the restart token file to the appropriate sources. PWXPC then assigns the restart tokens

in the state file or state tables to the specified sources, provided that the token values have not been supplied by the restart token file. For all remaining sources, PWXPC assigns the oldest restart point based on the available restart tokens.

- If state file or state table entry exists for all sources in the session, PWXPC assigns the restart tokens in the restart token file to the appropriate sources. For all remaining sources, PWXPC assigns the restart tokens from the state file or state tables.
- If the restart token file contains a special override statement only, PWXPC assigns the restart tokens in the special override statement to all sources.
- If the restart token file contains a special override statement and explicit override statements, PWXPC assigns the restart tokens in the explicit override statements to the specified sources. PWXPC then assigns the restart tokens in the special override statement to all remaining sources.

Default Restart Points

If you start a new CDC session and no restart token file exists, the PowerCenter Integration Service still runs the session. PWXPC passes null restart tokens for all of the sources to PowerExchange.

To indicate that the restart tokens are null, PWXPC issues message PWXPC_12060. PowerExchange then assigns the earliest possible restart point to each source.

Warning: Null restart tokens might produce unexpected results. Informatica recommends that you always specify valid restart tokens that indicate a point in the change stream where the source and target are in a consistent state.

The following table describes the default restart points that PowerExchange uses when it receives null restart tokens, by source and connection type:

Source Type	PWX CDC Change Connection ¹	PWX Real Time Connection ²
All z/OS data sources	The oldest PowerExchange Condense condense file that is recorded in the CDCT file.	The PowerExchange Logger for z/OS determines the best available restart point. Usually, this point is the oldest restart point for which an archive log is available, or if no archive log is available, the oldest restart point for an active log.
DB2 for i5/OS	The oldest PowerExchange Condense condense file that is recorded in the CDCT file.	The oldest journal receiver that is still on the journal receiver chain.
DB2 for Linux, UNIX, and Windows	The oldest PowerExchange Logger for Linux, UNIX, and Windows log file that is recorded in the CDCT file.	The current log position at the time the PowerExchange capture catalog was created.
Microsoft SQL Server	The oldest PowerExchange Logger for Linux, UNIX, and Windows log file that is recorded in the CDCT file.	The oldest data that is available in the Publication database.
Oracle	The oldest PowerExchange Logger for Linux, UNIX, and Windows log file that is recorded in the CDCT file - for both PowerExchange Express CDC for Oracle and PowerExchange Oracle CDC with LogMiner.	The earliest available point in the change stream: <ul style="list-style-type: none"> - For PowerExchange Oracle CDC with LogMiner, the most recent Oracle catalog dump in the archive logs. - For PowerExchange Express CDC for Oracle, the beginning of the most recent archive log.

Source Type	PWX CDC Change Connection ¹	PWX Real Time Connection ²
Any data source type that is logged to a PowerExchange Logger for Linux, UNIX, and Windows ³	The oldest PowerExchange Logger for Linux, UNIX, and Windows log file that is recorded in the CDCT file.	The oldest PowerExchange Logger for Linux, UNIX, and Windows log file that is recorded in the CDCT file.
<p>1. With a PWX CDC Change connection, you must use batch extraction mode.</p> <p>2. With a PWX CDC Real Time connection, use real time extraction mode to extract data from the source change stream or use continuous extraction mode to extract data that is logged to a PowerExchange Logger for Linux, UNIX, and Windows instance.</p> <p>3. Includes Linux, UNIX, and Windows data sources that are logged to a local or remote PowerExchange Logger for Linux, UNIX, and Windows instance. Also includes z/OS and i5/OS data sources that are logged to a remote PowerExchange Logger for Linux, UNIX, and Windows instance.</p>		

PowerExchange uses the default restart point only if all sources in a CDC session have null restart tokens. If some sources have non-null restart tokens, PWXPC assigns the oldest restart point of those restart tokens to sources for which no restart tokens are specified.

For example, a new CDC session contains the sources A, B, and C. The restart token file contains restart tokens for sources A and B. The restart point for source A is older than that for source B. Source C does not have existing or supplied restart tokens. Because some sources in the CDC session have explicit restart points, PWXPC does not assign null restart tokens to source C. Instead, PWXPC assigns the restart point for source A to source C because this restart point is the oldest one supplied.

RELATED TOPICS:

- [“Previewing PowerExchange Change Data” on page 46](#)

Initializing and Running CDC Sessions

After the restart information for every source is determined, PWXPC:

- Flushes the restart tokens to the state tables for relational targets and to the state file for nonrelational targets
- Writes an empty restart token file
- Creates the initialization restart token file containing the reconciled restart information

PWXPC passes the restart tokens for all sources to PowerExchange. PowerExchange uses the oldest restart token passed by PWXPC to start extracting data from the change stream. PowerExchange does not pass data for a source until its restart point is reached. This prevents targets from being updated with records processed in previous extraction runs.

PWXPC continually updates the restart tokens for each source in the state table or the state file as it issues flushes target data. With relational target tables in the same database, the PowerCenter Integration Service updates both the target tables and the restart tokens within a single commit. The PowerCenter Integration Service does separate commits for each unique relational database. With heterogeneous targets, the restart tokens in one relational database may differ from those in another relational database at specific points in time.

When you use nonrelational targets, the state file and the targets likely exist on completely different machines. With nonrelational targets, the PowerCenter Integration Service updates the targets and the state file in separate operations. If the session fails after the PowerCenter Integration Service commits data to the target but before it updates the restart tokens in the state file, targets may receive duplicate data when restarted. On warm start, PWXPC uses the last restart tokens written prior to the failure. As a result, PWXPC re-sends data which has already been applied to the nonrelational targets.

Ending CDC Sessions

Stop CDC sessions with PowerCenter and PowerExchange commands. Sessions can also fail or terminate unexpectedly.

When you stop a CDC session, PWXPC flushes any complete units of work that remain in its buffer. PWXPC then waits for confirmation from the PowerCenter Integration Service that the flushed data has been written to the targets. If it has, PWXPC writes the termination restart token file with the final restart tokens and issues the following message:

```
PWXPC 12075 [INFO] [CDCRestart] Session complete. Next session will restart at: :  
Restart 1 [restart1] : Restart 2 [restart2]
```

The PowerCenter Integration Service commits the flushed data to the targets, including the restart tokens for relational targets. After the PowerCenter Integration Service writes the flushed data to any nonrelational targets, it updates the state file with the restart tokens.

If the session fails, the PowerCenter Integration Service rolls back any uncommitted data and the related restart tokens for relational targets. This leaves only the last successfully committed UOW data and restart tokens in the relational target tables. The PowerCenter Integration Service uses relational database rollback capabilities to ensure that uncommitted data is removed during session termination. Consistency between the restart tokens and the relational target data is guaranteed because they are both committed within the same commit scope.

The PowerCenter Integration Service does not do rollback processing for nonrelational targets. As a result, duplicate data can occur on restart. You should account for this in your CDC session design.

Tip: To avoid the possibility of duplicate data, design the CDC sessions in your application to use relational targets only.

RELATED TOPICS:

- [“Stopping CDC Sessions” on page 178](#)

Creating Recovery Tables

The PowerCenter Integration Service creates the recovery tables if they do not exist. The PowerCenter Integration Service uses the database user name specified in the connection to create the tables. The database user name PowerExchange uses to create the recovery tables varies based on the PWXPC connection.

Note: The PowerCenter Integration Service does not create recovery queues. You must create the recovery queue manually.

Recovery Table Creation with PowerExchange Targets

PowerExchange create recovery tables for PWX DB2zOS connections.

PWX DB2zOS Relational Connections for DB2 for z/OS Targets

When PowerExchange creates the recovery tables for PWX DB2zOS connections, it uses:

- The default DB2 database (DSNDB04).
- The PowerExchange Listener user ID if the PowerExchange Listener is running with SECURITY=0 or SECURITY=1. This user must be granted the appropriate table creation privilege.

- The PowerExchange Listener user ID if the PowerExchange Listener is running with SECURITY=2 and MVSDB2AF=CAF. This user must be granted the appropriate table creation privilege.
- The database user name in the target connection if the PowerExchange Listener is running with SECURITY=2 and MVSDB2AF=RRSAF. This user must be granted the appropriate table creation privilege.

PWX DB2i50S Relational Connections for DB2 for i5/OS Targets

When PowerExchange creates the recovery tables for PWX DB2i50S connections, it uses:

- The PowerExchange Listener user ID if it is running with SECURITY=0 or SECURITY=1. This user must be granted the appropriate table creation privilege.
- The database user name in the target connection is if the PowerExchange Listener is running with SECURITY=2. This user must be granted the appropriate table creation privilege.
- The default journal so it must be enabled for the user name.

PWX DB2LUW Relational Connections for DB2 for Linux, UNIX, and Windows Targets

When PowerExchange creates the recovery tables for PWX DB2LUW connections, it uses:

- The default table space for user-defined tables.
- The database user name in the target connection. This user must be granted the appropriate table creation privilege.

PWX MSSQLServer Relational Connections for Microsoft SQL Server Targets

When PowerExchange creates the recovery tables for PWX MSSQLServer connections, it uses the database user name in the target connection. This user must be granted the appropriate table creation privilege.

PWX Oracle Relational Connections for Oracle Targets

When PowerExchange creates the recovery tables for PWX Oracle connections, it uses the database user name in the target connection. This user must be granted the appropriate table creation privilege.

Creating the Recovery Tables Manually

Consider manually creating the recovery tables so you can control attributes such as the database, table space, and buffer pools. Informatica provides SQL scripts in the following directory:

```
<PowerCenter_installation_directory>\server\bin\RecoverySQL
```

The following table lists the scripts that you can run to create the recovery tables in the target database:

Script	Database
create_schema_db2.sql	DB2
create_schema_ora.sql	Oracle
create_schema_sql.sql	SQL Server

The scripts are generic DDL. Make the appropriate changes for your environment.

Configuring the Restart Token File

You can configure the restart token file to specify the point from which you want to extract source data. If you do not know the restart token file name and location, use one of the following options:

- Look at the PWXPC_12057 message in the session log. PWXPC includes the restart token file folder and the restart token file name in this message.
- Open the application connection associated with the source. The application connection contains the restart token file name and folder location. This file name overrides the file name you specified in the application connection.
- If the restart token file name is not specified in the application connection, PWXPC uses the application name, if specified. Otherwise, PWXPC uses the workflow name.

Warning: The Restart Token File Name must be unique for every session. Non-unique names cause unpredictable results including session failures and potential data loss.

Restart Token File Control Statement Syntax

In the restart token file, you can specify explicit override statements, special override statements, and comments.

For explicit override control statements, use the following syntax:

```
extraction_map_name={sequence_token|CURRENT_RESTART}  
extraction_map_name={restart_token|CURRENT_RESTART}
```

For special override control statements, use the following syntax:

```
RESTART1={sequence_token|CURRENT_RESTART}  
RESTART2={restart_token|CURRENT_RESTART}
```

For comments, use the following syntax:

```
<!-- comments
```

The following rules and guidelines apply:

- Control statements can begin in any column.
- All control statements are optional.
- Do not include blank lines between control statements.
- Comment lines must begin with:

```
<!--
```
- You can specify both explicit override statements and one special override statement.
- An explicit override statement for a source takes precedence over any special override statement.
- On warm start, explicit override statements and the special override statement take precedence over any restart token values stored for sources in the start table or file.

RELATED TOPICS:

- [“How Restart Points Are Determined” on page 163](#)

Restart Token File Control Statement Parameters

You can specify control statements in the restart token file.

Comment Statement

You can use the comment statement anywhere in the restart token file. Comment statements must begin with “<!--”.

Explicit Override Statement

The explicit override control statement defines the restart token pair for a specific source.

A source can have multiple extraction maps, each with a different name.

You can use explicit override statements with a special override statement. The explicit override statements provide restart tokens for specific sources, and the special override statement provides restart tokens for any other sources.

The explicit override statement has the following syntax:

```
extraction_map_name={restart1_token|CURRENT_RESTART}  
extraction_map_name={restart2_token|CURRENT_RESTART}
```

The parameters are:

extraction_map_name

An extraction map name that matches a source in the CDC session. To determine the extraction map name, check one of the following:

- The **Extraction Map Name** attribute in the session properties for relational sources.
- For CDC data map sources, the **Schema Name Override** and **Map Name Override** attributes in the session properties. These attributes override the schema and map names in the extraction map for the source.
- For CDC data map sources, the **Schema Name** and **Map Name** values in the source Metadata Extensions in Designer.

restart1_token

The sequence token. The token format varies based on the source type. You can get the token value in the following ways:

- Use the PRINT statement of the DTLUAPPL utility to get a hexadecimal sequence value and then add eight zeroes to the end of that value.
- View the PowerExchange-generated DTL__CAPXRESTART1 field in extracted change records.
- Look for the sequence token in PowerExchange messages such as PWX-04565 and PWX-09959.
- Look for Restart Token 1 in PWXPC messages such as PWXPC_12060 and PWXPC_12068.

restart2_token

The restart token. The token format varies based on the source type. You can get the token value in the following ways:

- Use the PRINT statement of the DTLUAPPL utility to get the Restart value.
- View the PowerExchange-generated DTL__CAPXRESTART2 field in extracted change records.
- Look for the Logger token in PowerExchange messages such as PWX-04565 and PWX-09959.
- Look for Restart Token 2 in PWXPC messages such as PWXPC_12060 and PWXPC_12068.

CURRENT_RESTART

Generates a current restart token. The CDC reader opens a separate connection to PowerExchange to request generation of current restart tokens and then provides the generated restart tokens to all

applicable sources. Do not specify both CURRENT_RESTART and specific sequence and restart token values with an extraction name.

Note: You can use CURRENT_RESTART only for real-time connections.

You can also generate current restart tokens in the **Database Row Test** dialog box in the PowerExchange Navigator.

Special Override Statement

Use a special override statement to define restart tokens for one or more sources.

You can use a special override statement with explicit override statements. The special override statement then provides restart tokens for the sources without explicit override statements or an entry in the state table or file.

A special override statement must include both the RESTART1 and RESTART2 parameters:

RESTART1={*restart1_token* | CURRENT_RESTART}

Specifies the sequence token. The token format varies based on the source type. Enter one of the following keywords:

restart1_token

A specific sequence token. You can get the token value in the following ways:

- Use the PRINT statement of the DTLUAPPL utility to a hexadecimal sequence value and then add eight zeroes at the end.
- View the PowerExchange-generated DTL__CAPXRESTART1 field in extracted change records.
- Look for the sequence token in PowerExchange messages such as PWX-04565 and PWX-09959.
- Look for Restart Token 1 in PWXPC messages such as PWXPC_12060 and PWXPC_12068.

CURRENT_RESTART

PowerExchange generates a current restart token. The CDC reader opens a separate connection to PowerExchange to request generation of current restart tokens and then provides the generated restart tokens to all applicable sources.

Note: You can use CURRENT_RESTART only for real-time connections. You can also generate current restart tokens in the **Database Row Test** dialog box in the PowerExchange Navigator.

RESTART2={*restart2_token* | CURRENT_RESTART}

Specifies the restart token. The token format varies based on the source type. Enter one of the following keywords:

restart2_token

A specific restart token. You can get the token value in the following ways:

- Use the PRINT statement of the DTLUAPPL utility to get the Restart value.
- View the PowerExchange-generated DTL__CAPXRESTART2 field in extracted change records.
- Look for the Logger token in PowerExchange messages such as PWX-04565 and PWX-09959.
- Look for the Restart Token 2 in PWXPC messages such as PWXPC_12060 and PWXPC_12068.

CURRENT_RESTART

Generates a current restart token. The CDC reader opens a separate connection to PowerExchange to request generation of current restart tokens and then provides the generated restart tokens to all applicable sources.

Note: You can use CURRENT_RESTART only for real-time connections.

You can also generate current restart tokens in the **Database Row Test** dialog box in the PowerExchange Navigator.

Restart Token File Example

An example restart token file demonstrates the use of explicit and special override statements to provide restart tokens for a CDC session with seven source tables.

The example restart token file contains the following statements:

```
<!-- Restart Tokens for existing tables -->
restart1=000000AD775600000000000000AD77560000000000000000
Restart2=C1E4E2D3404000000AD5F2C00000000
<!-- Restart Tokens for the Table: rrtb0001_RRTB_SRC_001 -->
dldsn7.rrtb0001_RRTB_SRC_001=0000060D1DB2000000000000060D1DB20000000000000000
dldsn7.rrtb0001_RRTB_SRC_001=C1E4E2D340400000013FF362000000000
<!-- Restart Tokens for the Table: rrtb0001_RRTB_SRC_002 -->
dldsn7.rrtb0002_RRTB_SRC_002=000000A3719500000000000000A3719500000000000000000
dldsn7.rrtb0002_RRTB_SRC_002=C1E4E2D34040000000968FC6000000000
<!-- Restart Tokens for the Table: rrtb0001_RRTB_SRC_004 -->
dldsn7.rrtb0004_RRTB_SRC_004=000006D84E7800000000000006D84E7800000000000000000
dldsn7.rrtb0004_RRTB_SRC_004=C1E4E2D340400000060D1E61000000000
```

This file contains explicit override statements for three sources: RRTB_SRC_001, RRTB_SRC_002, and RRTB_SRC_004. The file also contains a special override statement for the remainder of the sources in the session. When the session runs, PWXPC writes message PWXPC_12060 to the session log. The message contains the following information:

```
=====
Session restart information:
=====
Extraction Map Name      Restart Token 1      Restart Token 2      Source
dldsn7.rrtb0001_RRTB_SRC_001 0000060D1DB2000000000000060D1DB20000000000000000 C1E4E2D340400000013FF362000000000 Restart file
dldsn7.rrtb0002_RRTB_SRC_002 000000A371950000000000000A371950000000000000000 C1E4E2D34040000000968FC6000000000 Restart file
dldsn7.rrtb0003_RRTB_SRC_003 000000AD77560000000000000AD77560000000000000000 C1E4E2D3404000000AD5F2C0000000000 Restart file (special override)
dldsn7.rrtb0004_RRTB_SRC_004 000006D84E780000000000006D84E78000000000000000 C1E4E2D340400000060D1E61000000000 Restart file
dldsn7.rrtb0005_RRTB_SRC_005 000000AD77560000000000000AD77560000000000000000 C1E4E2D3404000000AD5F2C0000000000 Restart file (special override)
dldsn7.rrtb0006_RRTB_SRC_006 000000AD77560000000000000AD77560000000000000000 C1E4E2D3404000000AD5F2C0000000000 Restart file (special override)
dldsn7.rrtb0007_RRTB_SRC_007 000000AD77560000000000000AD77560000000000000000 C1E4E2D3404000000AD5F2C0000000000 Restart file (special override)
```

For sources that have explicit overrides, the message displays "Restart file" in the Source column. For sources that have the special override restart tokens, the message displays "(special override)."

Using PWXPC Restart and Recovery

For CDC sessions, you must select the recovery strategy of **Resume from the last checkpoint** in the session properties to use PowerCenter and PowerExchange restart and recovery functionality. With this recovery strategy, the PowerCenter Integration Service can recover target table data and PWXPC can recover the CDC restart information.

PowerExchange supplies CDC restart tokens that describe a specific start point in the change stream. For relational targets, PWXPC stores the restart tokens in the recovery state tables in the target databases. For nonrelational targets, PWXPC stores the restart tokens in the recovery state file on the PowerCenter Integration Service machine.

PWXPC issues the following message when the **Resume from last checkpoint** recovery strategy is enabled:

```
PWXPC_12094 [process_id] Advanced GMD recovery in effect. Recovery is automatic.
```

With this recovery strategy, after a failure, PWXPC can automatically recover sessions that are warm started.

If you use the **Resume from the last checkpoint** recovery strategy, the following considerations apply:

- If you have a PowerCenter high availability (HA) system, you must enable HA recovery for your CDC workflows. Otherwise, when you warm start a CDC session, a restart error occurs and you must cold start the session. To configure HA recovery for a CDC workflow and avoid this error, in Workflow Manager, navigate to the **Properties** tab for the workflow and then select the **Enable HA Recovery** option.
- If you use any of the following transformation types for your sessions, you must select the **Output is Deterministic** option for the transformation:
 - External Procedure transformation
 - HTTP transformation
 - Java transformation
 - Source Qualifier transformation
 - SQL transformation
 - Stored Procedure transformation
 - Unstructured Data transformation

If you use a Lookup transformation, select either the **Lookup Source is Static** or **Lookup Cache Persistent** option for the transformation instead.

If you do not select the **Output is Deterministic** option or the **Lookup Source is Static** or **Lookup Cache Persistent** option for the transformation when using the **Resume from last checkpoint** recovery strategy, you cannot validate the CDC session in Workflow Manager.

- If you use the File Writer to write CDC data to flat files and other target types, such as relational targets, in the same session, do not enable resume-from-last-checkpoint recovery processing. If you do, the restart tokens might be compromised, causing possible data loss or duplication.
- If you run a session with **Resume from last checkpoint** recovery strategy and the session fails, do not alter the mapping, session, or state information before you restart the session. PWXPC cannot guarantee recovery if you make any of these changes.
- When the PowerCenter Integration Service resumes a session, it restores the session state of operation, including the state of each source, target, and transformation. PWXPC, in conjunction with the PowerCenter Integration Service, determines how much of the source data it needs to reprocess.

RELATED TOPICS:

- [“Understanding PWXPC Restart and Recovery” on page 160](#)

Enabling Recovery for a CDC Session

1. In the session properties, select **Resume from the last checkpoint** for the **Recovery Strategy** property. This recovery strategy is required for the PWXPC and PowerCenter Integration Service to recover CDC sessions.
2. For the **Commit Type** property, enter **Source**. CDC sessions always use source-based commit processing regardless of the **Commit Type** setting.

3. Clear the **Commit at End of File** property.

After PWXPC shuts down, the PowerCenter Integration Service issues a commit when the session ends. Data that is written to the targets after PWXPC shuts down is not reflected in the restart tokens. Disable this property to ensure that PWXPC issues all commits.

RELATED TOPICS:

- [“Recovering from CDC Session Failures” on page 180](#)

Configuring CDC Sessions for Restart Processing

Control PWXPC restart processing with session options and attributes.

The following table summarizes the options and attributes with defaults that must be changed for CDC sessions:

Option or Attribute	Location	Recommended Value and Default
Commit Type	Properties Tab	Specify Source . The default value is Target. The PowerCenter Integration Service automatically overrides the default to Source. Unless this value is Source, you cannot change Commit On End Of File .
Commit On End Of File	Properties Tab	Disable this option. By default, this option is enabled and PowerCenter Integration Service performs a commit when the session ends. This commit occurs after PWXPC commits the restart tokens. As a result, duplicate data can occur on restart.
Recovery Strategy	Properties Tab	Select Resume from the last checkpoint . PWXPC requires this setting for recovery. If you use PowerCenter high availability (HA) option, you must also specify the Enable HA Recovery option for the workflow. If you use a transformation type for which the Output is Deterministic option is available, ensure that you select this option. If you use a Lookup transformation, select the Lookup Source is Static or Lookup Cache Persistent option instead. If you do not select one of these options, you cannot validate the CDC session in Workflow Manager. The default value is Fail task and continue workflow .
Target Load Type	Mapping Tab, Targets Node	Select Normal . The default value is Bulk . To use recovery, you must select Normal.
Stop on errors	Config Object Tab	Enter 1 to ensure target data and restart token integrity are maintained. The default value is 0. With the default, any errors that the PowerCenter Integration Service encounters while writing to the target database or target files are not counted. Writer threshold errors can include key constraint violations, loading nulls into a not null field, and database trigger responses. In this case, PWXPC does not detect that the writer has not applied the data to the target and consequently advances the restart tokens.
Application Name	Application Connection	Specify a unique name. The default value is the first 20 characters of the workflow name. However, the default might not result in a unique name.

Option or Attribute	Location	Recommended Value and Default
Restart Token File Name	Application Connection	Specify a unique name. The default is either the application name, or the workflow name if no application name is specified. However, the default might not result in a unique name.
Number of Runs to Keep RestartToken File	Application Connection	Specify a value greater than 0. The default is 0. With the default, PWXPC keeps only one backup copy of the restart token initialization and termination files. Specify a value greater than 0 to retain history for recovery purposes.

RELATED TOPICS:

- [“Restart Token File” on page 162](#)
- [“Application Name Requirement for Restart Processing” on page 175](#)
- [“Enabling Recovery for a CDC Session” on page 173](#)

Application Name Requirement for Restart Processing

You must specify a unique application name for each CDC session to ensure proper restart processing.

Warning: if you use the same application name for different CDC sessions, data integrity might be damaged or data loss might occur.

Generating Current Restart Tokens for CDC Sessions

Use one of the following methods to generate current restart tokens for CDC data sources:

- **The special override statement with CURRENT_RESTART in the PWXPC restart token file.** PWXPC and PowerExchange generate restart tokens that represent the current end of the change stream at the time the PowerCenter session runs.
- **The DTLUAPPL utility with the GENERATE RSTTKN parameters and a valid capture registration.** DTLUAPPL generates restart tokens that represent the current end of the change stream at the time the utility runs.
- **PowerExchange Navigator database row test function.** PowerExchange generates restart tokens with the location and source type you specify. These restart tokens represent the current end of the change stream at the time at which you perform the database row test.

RELATED TOPICS:

- [“Configuring the Restart Token File” on page 169](#)

DTLUAPPL with CDC Sessions

You can use the PowerExchange DTLUAPPL utility to generate current restart tokens for CDC data sources. You can then update the restart token file with the generated restart tokens to establish the starting extraction point in the change stream for a CDC session.

Example of Generating Restart Tokens with DTLUAPPL

The registration on the ADD RSTTKN statement must exist. This registration must be associated with the instance name specified on the MOD APPL statement. You can use either ADD APPL or MOD APPL for a new application name. DTLUAPPL creates the application name if it does not exist when MOD APPL is used.

Tip: Select an application name and use it only with DTLUAPPL when generating restart tokens.

The following example generates restart tokens for source registration DB2DEMO1 with an application name of *tokens* and then prints those restart tokens:

```
MOD APPL tokens DSN7 RSTTKN GENERATE
  ADD RSTTKN db2demo1
END APPL tokens
PRINT APPL tokens
```

The DTLUAPPL prints the generated tokens because the PRINT APPL statement is specified:

```
Application name=<tokens> Rsttkn=<1> Ainseq=<0> Preconfig=<N>
  FirstTkn   =<>
  LastTkn    =<>
  CurrentTkn =<>
Registration name=<db2demo1.1> tag=<DB2DSN7db2demo11>
  Sequence=<000007248B9600000000000007248B9600000000>
  Restart  =<D2D1D4D34040000007248B0E00000000>
```

DTLUAPPL does not generate the complete *restart1_token* value which is shown in the SEQUENCE token. You must add the trailing four bytes (eight digits) of zeros manually when you update the restart token file. DTLUAPPL does generate the complete *restart_token2* value in the Restart= token. You can copy this value to the restart token file.

Tip: You can use same restart tokens for multiple source tables in the restart token file to start extracting changes from the same point in the change stream. You only need to run DTLUAPPL multiple times if you want to start extracting changes from different locations in the change stream for different sources.

By using the tokens in this example, the restart token file looks as follows:

```
D1DSN7.db2demo1=000007248B9600000000000007248B9600000000000000
D1DSN7.db2demo1=D2D1D4D34040000007248B0E00000000
```

Starting CDC Sessions

You can use the Workflow Manager, Workflow Monitor, or *pmcmd* program to start a workflow or task.

You can run the entire workflow, a part of the workflow, or a task in the workflow.

You can warm start, cold start, or recover workflows and tasks.

Cold Start Processing

Cold start workflows and tasks by selecting the cold start command in Workflow Manager, Workflow Monitor, or **pmcmd**. When you request cold start for CDC sessions, PWXPC writes the following message in the session log:

```
PWXPC_12091 [INFO] [CDCRestart] Cold start requested
```

PWXPC performs the following tasks for cold start:

1. Reads the restart tokens from the restart token file only.
2. Commits the restart tokens to the state tables and files, and then writes message PWXPC_12104 to the session log.
3. Continues processing and committing the change data and restart tokens, until the session ends or is stopped.

RELATED TOPICS:

- [“How Restart Points Are Determined” on page 163](#)

Warm Start Processing

Warm start workflows and tasks by selecting the start or restart command in Workflow Manager, Workflow Monitor, or *pmcmd*. When you request warm start, PWXPC writes the following message to the session log:

```
PWXPC_12092 [INFO] [CDCRestart] Warm start requested. Targets will be resynchronized automatically if required
```

PWXPC automatically performs recovery when you warm start a workflow or task. You do not need to recover workflows and tasks before you restart them.

PWXPC performs the following tasks for warm start:

1. Reconciles the restart tokens from the restart token file and from the recovery state tables and files.
2. For heterogeneous targets, PWXPC queries the PowerCenter Integration Service about the commit levels of all targets. If all targets in the session are at the same commit level, PWXPC skips recovery processing.
3. If recovery is required for heterogeneous targets, PWXPC re-reads the data for the last UOW committed to higher-level targets and flushes it to those targets with the lower commit level. The PowerCenter Integration Service commits any flushed data and restart tokens to any relational targets and updates any nonrelational files.
Note: PWXPC does not read the restart token file if recovery is required.
4. If recovery is not required and the reconciled restart tokens differ from those in the state tables and files, PWXPC commits the reconciled restart tokens and issues message PWXPC_12104.
5. Continues processing and committing data and restart tokens until the session ends or is stopped.

RELATED TOPICS:

- [“How Restart Points Are Determined” on page 163](#)

Recovery Processing

Recover workflows and tasks by selecting the recover command in Workflow Manager, Workflow Monitor, or *pmcmd*. When you request recovery, PWXPC writes the following message to the session log:

```
PWXPC_12093 [INFO] [CDCRestart] Recovery run requested. Targets will be resynchronized if required and processing will terminate
```

Select Recover to populate the restart token file with the restart tokens for all sources in the CDC session so that you can cold start. You can also use recovery to ensure the targets and restart tokens are in a consistent state.

PWXPC automatically performs recovery when you warm start a workflow or task. You do not need to recover workflows and tasks before you restart them. PWXPC performs the following tasks for recovery:

1. Reads the restart tokens from the recovery state tables and file.
2. Creates the initialization restart token file with the reconciled restart tokens.
3. For heterogeneous targets, PWXPC queries the PowerCenter Integration Service about the commit levels of all targets. If all targets in the session are at the same commit level, PWXPC skips recovery processing.
4. If recovery is required for heterogeneous targets, PWXPC re-reads the data for the last UOW committed to higher-level targets and flushes it to those targets with the lower commit level. The PowerCenter

Integration Service commits any flushed data and restart tokens to any relational targets and updates any nonrelational files.

Note: PWXPC does not read the restart token file if recovery is required.

5. Updates the restart token file with the final restart tokens, creates the termination restart token file, and ends.

You can now warm start or cold start the workflow or task to process change data from the point of interruption.

Stopping CDC Sessions

Stop CDC sessions through PowerCenter by issuing the stop command from Workflow Monitor or **pmcmd**. You can also stop CDC sessions through PowerExchange by issuing the STOPTASK command or by running the DTLUTSK utility. You can also abnormally terminate a workflow or task with the abort command in Workflow Monitor or **pmcmd**.

When you stop a workflow or task gracefully by issuing the stop command through PowerCenter or PowerExchange, the following actions occur:

1. If you issue the PowerCenter stop command, the PowerCenter Integration Service requests PWXPC to stop.
If you issue the PowerExchange stop command, PowerExchange sends an end of file to PWXPC.
2. PWXPC performs end of file processing to flush the remaining uncommitted complete units of work and their associated restart tokens to the targets. PWXPC then writes messages PWXPC_12101 and PWXPC_12068 to the session log.
3. The PowerCenter Integration Service processes all of data in the pipeline and writes it to the targets.
4. The PowerCenter Integration Service sends an acknowledgement to PWXPC indicating that the targets have been updated.
5. PWXPC writes the termination restart token file, and then writes message PWXPC_12075 to the session log. After this, the PWXPC reader shuts down.
6. The PowerCenter Integration Service ends the session successfully.

You can also stop CDC sessions gracefully if you use:

- Idle Time=0 in the PWX CDC Real Time connection. Idle Time=0 instructs PowerExchange to stop processing at end of log.
- PWX CDC Change connection to extract changes from condense files. When you use PowerExchange batch extraction mode for condense files, the extraction automatically ends when all condensed data is read.

RELATED TOPICS:

- [“Configuring Idle Time” on page 116](#)

Changing CDC Sessions and Restarting from the Point of Interruption

You can add new sources and targets to an existing CDC sessions. However, you must cold start the session after you do this, which means you need to ensure you have the proper restart information for the original sources in the session.

To change a CDC session and restart it from the point of interruption:

1. Gracefully stop the workflow.
2. After the workflow stops successfully, issue the Recover command for the CDC session.
When you recover tasks, PWXPC writes the ending restart tokens for all sources in the session into the restart token file specified on the CDC connection.
3. Change the session or workflow as desired.
4. Ensure that the restart token file in the source CDC connection points to the same restart token file updated in the recovery session.
5. Optionally, add statements providing restart tokens to the restart token file for any new sources.
6. If you remove any sources from the session, update the restart token file to remove their restart tokens.
7. Cold start the CDC session.

RELATED TOPICS:

- [“Stopping CDC Sessions” on page 178](#)

Adding a Source to a Session - Example

In this example, a new source (RRTB_SRC_004) is added to an existing CDC session containing three sources. The restart points for the existing sources are maintained. DTLUAPPL is used to generate a restart token representing the current end-of-log in the change stream.

To add a new source to an existing session - example:

1. Stop the workflow by issuing the Stop command in Workflow Monitor.
2. After the workflow stops, use the Workflow Monitor and issue the Recover Task command from Workflow Monitor to run a recovery session. PWXPC writes the following messages in the session log:

```
PWXPC_12060 [INFO] [CDCRestart]
=====
Session restart information:
=====
Extraction Map Name      Restart Token 1      Restart Token 2      Source
dldsn7.rrtb0002_RRTB_SRC_002 000000AD220F00000000000000AD220F0000000000000000 C1E4E2D34040000000AD0D9C000000000 GMD storage
dldsn7.rrtb0001_RRTB_SRC_001 000000AD220F00000000000000AD220F0000000000000000 C1E4E2D34040000000AD0D9C000000000 GMD storage
dldsn7.rrtb0003_RRTB_SRC_003 000000AD220F00000000000000AD220F0000000000000000 C1E4E2D34040000000AD0D9C000000000 GMD storage
```

PWXPC places the restart tokens in the restart token file specified in the CDC application connection.

3. Make any necessary changes to the mapping, session, and workflow to add the new source, RRTB_SRC_004.
4. Run DTLUAPPL with RSTTKN GENERATE to generate restart tokens for the current end-of-log. Use the following DTLUAPPL control cards to do this:

```
mod APPL dummy DSN7 rsttkn generate
  mod rsttkn rrtb004
end appl dummy
print appl dummy
```

The PRINT command produces the following output:

```
Registration name=<rrtb004.1> tag=<DB2DSN7rrtb0041>
Sequence=<00000DBF240A0000000000000DBF240A00000000>
Restart =<C1E4E2D3404000000DBF238200000000>
```

Add eight zeroes to the end of the *Sequence=* value to create the restart token file value.

5. Update the restart token file to add the new source and its tokens. The updated file looks as follows:

```
<!-- existing sources
dldsn7.rrtb0001_RRTB_SRC_001=000000AD220F00000000000000AD220F0000000000000000
dldsn7.rrtb0001_RRTB_SRC_001=C1E4E2D34040000000AD0D9C000000000
```

```

dldsn7.rrtb0002_RRTB_SRC_002=000000AD220F00000000000000AD220F0000000000000000
dldsn7.rrtb0002_RRTB_SRC_002=C1E4E2D34040000000AD0D9C00000000
dldsn7.rrtb0003_RRTB_SRC_003=000000AD220F00000000000000AD220F0000000000000000
dldsn7.rrtb0003_RRTB_SRC_003=C1E4E2D34040000000AD0D9C00000000
<!-- new source
dldsn7.rrtb0004_RRTB_SRC_004=000000DBF240A0000000000000DBF240A0000000000000000
dldsn7.rrtb0004_RRTB_SRC_004=C1E4E2D34040000000DBF238200000000

```

6. Cold start the session.

PWXPC passes these restart tokens to PowerExchange to recommence extracting changes from the change stream.

Note: This restart point is earlier than the one just generated for the new source. The new source does not receive any changes until the first change following its restart point is encountered.

RELATED TOPICS:

- [“Configuring the Restart Token File” on page 169](#)
- [“DTLUAPPL with CDC Sessions” on page 175](#)

Recovering from CDC Session Failures

If you run a session with a resume recovery strategy and the session fails, do not edit the state information or the mapping for the session before you restart the session.

CDC sessions fail for a number of reasons, such as:

- Permanent errors such as source or target data errors
- Transitory errors such as infrastructure problems, server crashes, and network availability issues

If the session fails because of transitory errors, restart the session after the source of the transitory error is corrected. PWXPC automatically recovers warm started sessions, if required although you can also run a recovery session.

Note: You cannot override the restart point if recovery processing is required. PWXPC does not read the restart token file if you warm start and recovery is required or if you run a recovery session.

CDC sessions also fail because of permanent errors, such as SQL failures or other database errors. You must correct permanent errors before restarting the CDC session. With some failures, you can correct the error and then restart the CDC session. In other cases, you need to re-materialize the target table from the source table before you recommence applying changes to it. If you re-materialize the target table, you need to provide restart tokens matching the new restart point in the change stream and the cold start the CDC session.

RELATED TOPICS:

- [“Recovery Processing” on page 177](#)

Example: Session Recovery

In this example, a CDC session with relational targets is aborted in the Workflow Monitor. The Restart Task command is issued from the Workflow Monitor for the CDC session. PWXPC issues the following message:

```

PWXPC_12092 [INFO] [CDCRestart] Warm start requested. Targets will be resynchronized
automatically if required

```

PWXPC automatically recovers when the session is warm started. PWXPC issues the following messages displaying the restart tokens found for the session and its sources:

```

PWXPC_12060 [INFO] [CDCRestart]
=====
Session restart information:
=====

```

Extraction Map Name	Restart Token 1	Restart Token 2	Source
dldsn8.rrtb0004_RRTB_SRC_004	00000FCA6584000000000000D2E004A00000000FFFFFFFF	C1E4E2D3404000000D21B1A500000000	GMD storage
dldsn8.rrtb0009_RRTB_SRC_009	00000FCA6584000000000000D2E004A00000000FFFFFFFF	C1E4E2D3404000000D21B1A500000000	GMD storage
dldsn8.rrtb0005_RRTB_SRC_005	00000FCA6584000000000000D2E004A00000000FFFFFFFF	C1E4E2D3404000000D21B1A500000000	GMD storage
dldsn8.rrtb0006_RRTB_SRC_006	00000FCA6584000000000000D2E004A00000000FFFFFFFF	C1E4E2D3404000000D21B1A500000000	GMD storage
dldsn8.rrtb0008_RRTB_SRC_008	00000FCA6584000000000000D2E004A00000000FFFFFFFF	C1E4E2D3404000000D21B1A500000000	GMD storage
dldsn8.rrtb0003_RRTB_SRC_003	00000FCA6584000000000000D2E004A00000000FFFFFFFF	C1E4E2D3404000000D21B1A500000000	GMD storage
dldsn8.rrtb0002_RRTB_SRC_002	00000FCA6584000000000000D2E004A00000000FFFFFFFF	C1E4E2D3404000000D21B1A500000000	GMD storage
dldsn8.rrtb0001_RRTB_SRC_001	00000FCA6584000000000000D2E004A00000000FFFFFFFF	C1E4E2D3404000000D21B1A500000000	GMD storage
dldsn8.rrtb0007_RRTB_SRC_007	00000FCA6584000000000000D2E004A00000000FFFFFFFF	C1E4E2D3404000000D21B1A500000000	GMD storage

PWXPC issues the PWXPC_12069 message when it detects that recovery is required. The 12069 message usually includes the begin-UOW (*from*) and end-UOW (*to*) restart tokens for the oldest uncommitted UOW that PWXPC re-reads during recovery. PWXPC stores end-UOW restart tokens in the state table and file unless sub-packet commit is used.

```
PWXPC_12069 [INFO] [CDCRestart] Running in recovery mode. Reader will resend the the oldest uncommitted UOW to resync targets:
from: Restart 1 [00000FCA6584000000000000D2E004A00000000FFFFFFFF] : Restart 2 [C1E4E2D3404000000D21B1A500000000]
to: Restart 1 [00000FCA6584000000000000D300D8000000000FFFFFFFF] : Restart 2 [C1E4E2D3404000000D21B1A500000000].
```

The *from* restart tokens are the same as those displayed in the PWXPC_12060 messages for all sources. This restart token represents the start point in the change stream for the oldest uncommitted UOW. The *to* restart tokens represent the end of the oldest uncommitted UOW. Since the application connection for this session specifies sub-packet commit, the Restart 2 value is the begin-UOW value in both cases. The Restart 1 values represent the start and end change records in the Restart 2 UOW.

PWXPC rereads the changes between the two restart token values in the 12069 message and issues a commit for the data and the restart tokens. The PowerCenter Integration Service writes the data to the target tables and the restart tokens to the state table.

PWXPC and the PowerCenter Integration Service continue to read and write data and restart tokens until the session ends or is stopped.

RELATED TOPICS:

- [“Configuring Maximum Rows per Commit” on page 120](#)

Managing Session Log and Restart Token File History

When you run CDC sessions, session log and restart token file history is often useful for recovery situations. You can use these historical files to find previous restart points when data needs to be re-extracted.

By default, the PowerCenter Integration Service keeps a single session log for each workflow or session. Change the **Save session log for these runs** option in either the session or Session Configuration properties to specify the number or runs to keep the session logs. You can also change the **Save session log by** option to keep session logs by timestamp instead of number of runs.

PWXPC keeps a single pair of initialization and termination restart token files. In the source CDC connection, specify a value other than 0 or 1 in the **Number of Runs to Keep RestartToken File** option to retain more than a single pair of files.

You can determine starting and ending restart points for each extraction by using historical copies of the restart token file. You need historical copies of the session logs to re-extract changes at any point in between the session start and end.

When PWXPC issues a real-time flush to commit data to the targets, it issues message PWXPC_10081. This message contains the restart tokens at that point in time:

```
PWXPC_10081 [INFO] [CDCDispatcher] raising real-time flush with restart tokens
[<restart1_token>], [<restart2_token>] <because UOW Count [<n>] is reached.> | <because
Real-time Flush Latency [<n>] occurred.>
```

To restart an extraction from a specific commit point, use the restart tokens in the appropriate 10081 message to populate the restart token file and cold start the CDC session. PWXPC passes the restart token file values to PowerExchange to extract the data from that point forward.

CHAPTER 7

Flexible Target Key Transformation

This chapter includes the following topics:

- [Flexible Target Key Transformation Overview, 182](#)
- [Configuring PowerExchange Extraction Maps, 183](#)
- [Configuring Mappings with Flexible Target Key Transformations, 184](#)
- [Configuring Flexible Target Key Transformations with Multiple Sources, 185](#)

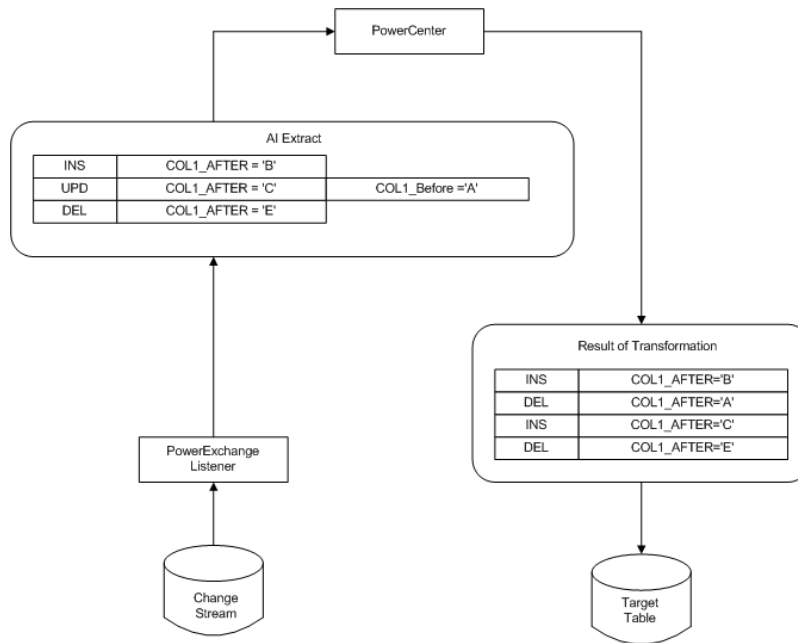
Flexible Target Key Transformation Overview

The Flexible Target Key Custom transformation has been developed to address the case where a data target has its Primary Key mapped to a column in a PowerExchange Change Data Capture source, which may be subject to update.

Prior to the implementation of the Flexible Target Key facility, the only way to identify these changes to a key field was to pass the full record before and after image in the change stream. For sessions involving large amounts of data this overhead could be significant.

For those wishing to keep network traffic to a minimum, PowerExchange has been enhanced to allow indicators to be added to specific fields. These indicators identify instances where that field has changed, and comprise both a “before image” field and a “change indicator” for any given field. From this information a delete/insert pair can then be generated for the target.

The following figure shows the process:



COL1 is updated from an A to a C. The PowerExchange Listener sends a before-image column value of A and an after-image column value of C to PowerCenter. Because COL1 is marked as the primary key for the target table, the Flexible Target Key transformation processes the update as a delete of row A and an insert of row C.

Configuring PowerExchange Extraction Maps

To ensure enough information is passed through to PowerCenter for a column, or columns, intended for use as a target primary key, a “change indicator” and “before image” field must be associated with the PowerExchange extraction map.

To configure a PowerExchange extraction map:

1. To add the before image and change indicator, right-click the extraction group.
2. Open the extraction group.
3. Open the extraction map. Either right-click the map and select **Open**, or double-click the extraction map.
4. Right-click the column that requires the before image and change indicator to be set.
5. Select **Amend Control Indicator/Before Image Extensions**.
6. To set the change indicator, select and add the required column by double-clicking or highlighting it and then clicking **Add**.
7. When all the required change indicators have been set up, click the **Before Images** tab.
8. Repeat the process for columns that require the before image to be included with the change capture data.

Note: The BI and CI column names can be changed on these screens by single-clicking and editing. The following sections in this chapter will refer to the default names.

Configuring Mappings with Flexible Target Key Transformations

The PowerExchange Change Data Capture (CDC) source will now need to be imported. This must be done through the Import from PowerExchange option under Source Analyzer in the PowerCenter Designer.

After the CDC source has been imported, the target definitions will be required. Import them in the Target Designer in the PowerCenter Designer.

The following figure shows an example structure of a source CDC table that works with a DB2 data capture source and a DB2 target table:

Name	Datatype
db2custc_CUSTOMER	
DTL_CAPXRES...	VARBIN
DTL_CAPXRES...	VARBIN
DTL_CAPXUOW	VARBIN
DTL_CAPXUSER	VARCHAR
DTL_CAPXTIME...	CHAR
DTL_CAPXACTI...	CHAR
CUSTOMER_ID	NUM32
SURNAME	CHAR
FIRST_NAME	CHAR
DATE_OF_BIRTH	DATE
CUST_START_D...	TIMESTAMP
PARTNER_CUST...	NUM32
DTL_CI_PARTN...	CHAR
DTL_BI_PARTN...	NUM32
PARTNER_SUR...	CHAR
PARTNER_FIRS...	CHAR

Note: The columns prefixed DTL_CI and DTL_BI are the change indicator and before image, respectively. These columns are necessary for this flexible transformation. The DTL__CAPX columns are PowerExchange capture columns.

The following figure shows the structure of the target DB2 table:

K.	Name	Datatype
	PARTNER_CU...	integer
	SURNAME	char
	FIRST_NAME	char
	CUSTOMER_ID	integer

The source table has a primary key of CUSTOMER_ID, and the PARTNER_CUST_ID is an attribute of the table. The transformation will load the data into the target table where the PARTNER_CUST_ID is the primary key, and the CUSTOMER_ID is an attribute of that table.

In this case, if the PARTNER_CUST_ID changes in the source table, a delete and insert will be required for the target.

Drag the PowerExchange CDC source, and the target into the mapping.

Note: Any of the control information including BI and CI columns could, if required, be defined in the target definition.

To configure a mapping with a Flexible Target Key transformations:

1. From the **Transformation** option, select **Create**.
2. Select a transformation type of **Flexible Target Key** and click **Create**.
The transformation is inserted into the mapping.
3. Link into the transformation the required columns from the source for the target.
Also, the transformation will require the BI and CI indicators that were assigned earlier in PowerExchange. A final column DTL__CAPXACTION should also be added to the transformation.
The BI and CI columns need to be linked to the Flexible transformation.
4. Right-click the transformation you have just created. Select **Edit**, and then select the **Source Column Map** tab.
5. Add the columns to which the relevant before images and change indicators are assigned.
Multiple columns can be added here when the target primary key demands.
6. Link the transformation to the target.
Only link the required data columns from the transformation to the target. No links will exist from the transformation to the target for the CI and BI columns, nor for the DTL__CAPXACTION column unless required in the target.
7. Create the workflow from the mapping as normal.

RELATED TOPICS:

- [“Working with Extraction Map Definitions” on page 40](#)
- [“Configuring PowerExchange Extraction Maps” on page 183](#)
- [“Source and Target Definitions” on page 25](#)

Configuring Flexible Target Key Transformations with Multiple Sources

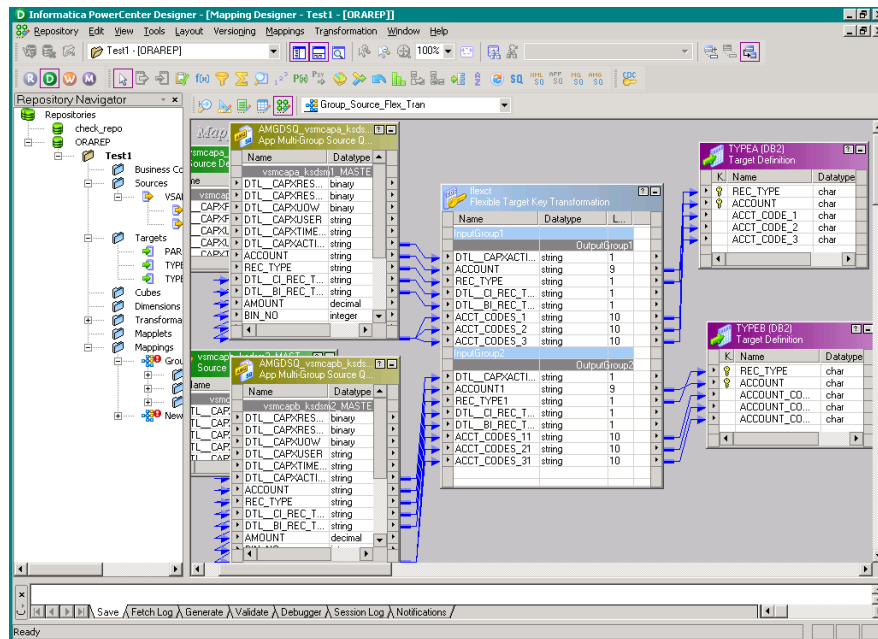
The following example shows how to process changes captured from a VSAM file with two different record types through a single Flexible Target Key transformation and then apply the changes to two DB2 target tables.

This example assumes that the data map for the VSAM data set has been modified to assign the relevant record id value test for both record types. In this example, a data map has been created for records with a REC_TYPE of “A” and for REC_TYPE of “B”. These data maps are called ksdss1.ksdsm1 and ksdss2.ksdsm2, respectively. These data maps have been registered for change data capture.

1. Assign the BI and CI PowerExchange fields to the extraction map.
2. Import the two extraction maps as data sources into the PowerCenter Designer by using the Import from PowerExchange dialog box.

Import the relevant data targets. In this example, the two input capture streams will be written to two separate DB2 tables.

The following figure shows the mapping:



The Flexible Target Key transformation contains the columns for both of the PowerExchange sources, and those sources each link to their own target tables. The Flexible Target Key transaction includes input and output groups. An input and output group is added when the transformation is created.

Subsequent input/output groups are required for each source. These are added when you “Edit” the transformation and select the Ports tab and then use the Create Input Group and Output Group buttons.

- Before validating the mapping, you must assign the before image and change indicators to the ports.

RELATED TOPICS:

- [“Configuring PowerExchange Extraction Maps” on page 183](#)

Part III: PowerExchange ODBC

This part contains the following chapters:

- [Installing PowerExchange ODBC, 188](#)
- [Working with Mappings for PowerExchange ODBC, 191](#)
- [Configuring Connections for PowerExchange ODBC, 195](#)
- [Working with Sessions for PowerExchange ODBC, 198](#)
- [PowerExchange Restart and Recovery, 202](#)

CHAPTER 8

Installing PowerExchange ODBC

This chapter includes the following topics:

- [Installing PowerExchange ODBC Overview, 188](#)
- [Modifying the PowerExchange Configuration Files, 188](#)
- [Creating ODBC Data Sources, 189](#)

Installing PowerExchange ODBC Overview

Before you use PowerExchange ODBC connections with PowerCenter, install and configure PowerExchange on the PowerCenter Client and PowerCenter Integration Service machines. Also, install the PowerExchange ODBC drivers on Windows.

After you install and configure PowerExchange, you can create PowerExchange ODBC data sources.

Note: When connecting to PowerExchange from PowerCenter, Informatica recommends that you use PWXPC instead of PowerExchange ODBC. PWXPC has additional functionality as well as improved performance and superior CDC recovery and restart.

For more information about PowerExchange ODBC, see the *PowerExchange Reference Manual*.

Modifying the PowerExchange Configuration Files

You must define nodes for the PowerExchange Listeners in the PowerExchange configuration file, `dbmover.cfg`, on the PowerCenter Integration Service and Client machines. The `NODE` statements specify the information that PowerExchange uses to communicate with Listeners that access the source or target databases or files.

Note: You can also use PowerExchange local mode if the data resides on the same machine as the PowerCenter Integration Service. In local mode, a PowerExchange Listener is not required. If you use local mode, specify **local** in the Location property or the `LOCATION` parameter for the ODBC data source. You do not need to update the PowerExchange `dbmover.cfg` file.

Local mode is not available on 32-bit Windows systems.

Creating ODBC Data Sources

Create ODBC data sources on the PowerCenter Client and PowerCenter Integration Service machines.

On the PowerCenter Client machine, create ODBC data sources so that PowerCenter can import the metadata for PowerExchange sources and targets and you can preview data from PowerCenter.

On the machine that runs the PowerCenter Integration Service, create ODBC data sources so that PowerCenter workflows can connect to the PowerExchange Listener and extract data for PowerExchange sources or targets.

Creating an ODBC Data Source on Windows

Create a PowerExchange ODBC data source on Windows for the PowerExchange Client and for the PowerExchange Integration Service if it runs on Windows. Use the **ODBC Data Source Administrator**.

1. To start the **ODBC Data Source Administrator**, perform one of the following actions:
 - If you plan to create a 32-bit data source on a Windows 32-bit system or a 64-bit data source on a Windows 64-bit system, open the **Control Panel** and click **Administrative Tools**. Then double-click **Data Sources (ODBC)**.
 - If you plan to create a 32-bit data source on a Windows 64-bit system, enter the following command at the command prompt:

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

Enter this command if you installed the PowerCenter Client, which is a 32-bit application, on a Windows 64-bit system.

The **ODBC Data Source Administrator** dialog box appears.

2. On the **System DSN** tab, click **Add**.
3. In the **Create New Data Source** wizard, select the Informatica PowerExchange driver from the list of available drivers and click **Finish**.

If this driver is not available, see the *PowerExchange Installation and Upgrade Guide*.

The **PowerExchange Data Source** dialog box appears.

4. On the **PowerExchange Data Source** tab, define the ODBC data source.
Enter values for all of the properties that are displayed for the selected data source type.
5. On the **General** tab, enter optional information that does not depend on the data source type.
6. On the appropriate **Properties** tab, enter properties that are specific to the data source type or access method.

The following table identifies the tabs on which to enter the information by data source or access type:

Data Source or Access Type	Properties Tab
CAPX	CAPX Properties tab
CAPXRT	CAPXRT Properties tab
DB2 for i5/OS	AS/400 Properties tab
DB2 for z/OS	DB2/S390 Bulk Load Properties tab

7. Click **OK**.

The ODBC data source appears in the **System Data Sources** list on the **System DSN** tab.

8. Click **OK**.

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

Creating a PowerExchange ODBC Data Source on Linux or UNIX

If the PowerCenter Integration Service runs on a Linux or UNIX machine, create a PowerExchange ODBC data source on that machine. To create the data source, update the appropriate `odbc.ini` file.

The data source entries in the `odbc.ini` file have the following format:

```
[data_source_name]
DRIVER=PowerExchange_full_install_path/libdtlodbc.sl
DESCRIPTION='descriptive_text'
LOCATION=data_source_node_from_dbmover.cfg
DBTYPE=access_method_for_file_or_database
... (other ODBC parameters as appropriate)
```

The data source name defined in the `odbc.ini` file is specified in the **Connect String** attribute of the ODBC connection in PowerCenter. This connect string is used to load the PowerExchange ODBC driver and connect to the specified location to extract or load data.

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

RELATED TOPICS:

- [“Creating an ODBC Data Source on Windows” on page 189](#)

CHAPTER 9

Working with Mappings for PowerExchange ODBC

This chapter includes the following topics:

- [Working with Mappings for PowerExchange ODBC Overview, 191](#)
- [Working with Source and Target Definitions for PowerExchange Batch \(ODBC\), 192](#)
- [Working with Source Definitions for PowerExchange Change or Real-time \(ODBC\), 193](#)

Working with Mappings for PowerExchange ODBC Overview

A mapping is a set of source and target definitions linked by transformation objects that define the rules for data transformation. Mappings represent the data flow between sources and targets.

Source and target definitions represent metadata for sources and targets. When you create a source definition, its structure differs depending on the type of source it represents:

- Nonrelational sources require a multi-group source definition.
- Relational sources use a single-group source definition.

The source qualifier for a source definition also differs in structure depending on the type of source definition.

After you create a source or target definition, you can include it in a mapping to extract data from the source or load data to the target. You can extract source data in batch, change, or real-time mode. You can use one source definition and one mapping for all modes.

RELATED TOPICS:

- [“PowerExchange ODBC Drivers” on page 18](#)

Working with Source and Target Definitions for PowerExchange Batch (ODBC)

With PowerExchange ODBC, you can create source or target definitions to access VSAM and sequential data sets (flat files), Adabas, DataCom, DB2, IDMS, or IMS databases. You create the source and target definitions identically regardless of the database type. After you create a source or target definition, you can edit it.

For nonrelational sources, the metadata imported is from the PowerExchange data map created through the PowerExchange Navigator. In the case of DB2 for z/OS and DB2 for i5/OS, the metadata is obtained either directly from the DB2 catalog or from a PowerExchange data map.

Importing a Source or Target Definition for PowerExchange ODBC

You can use ODBC to connect to a PowerExchange Listener to import a source or target definition. The data map must be sent to the PowerExchange Listener before you can import the data map by using an ODBC connection in Designer.

When you connect to the PowerExchange Listener, the Designer displays a list of schemas and tables. Before you connect to the source or target, you can filter the list by schema name. Then select a table from the list to create the source or target definition.

When you import a DB2 for z/OS or DB2 for i5/OS source definition through a PowerExchange Listener, the Designer imports the primary keys.

1. In the Source Analyzer, click **Sources > Import from Database** to import a source definition. In the Target Designer, click **Targets > Import from Database** to import a target definition.

The **Import Tables** dialog box appears.

2. In the **Owner name** field, you can enter filter criteria to narrow the list of objects that are retrieved.

When you import PowerExchange data maps for nonrelational sources or targets, ensure that the owner name is the schema name from the data map and the ODBC data source has a DB Type of NRDB or NRDB2. When the DB2 catalog is used for a DB2 for z/OS or DB2 for i5/OS source or target, ensure that the owner is the owner of the DB2 tables and the DB Type is either DB2 or DB2400C.

Note: The only difference between NRDB and NRDB2 is whether a three- or two-tier naming format is used for tables in the SQL statements for extracting or loading data:

schema.mapname.table for NRDB

schema.mapname_table for NRDB2

If the ODBC data source or target points to a PowerExchange Listener that is running with a PowerExchange security setting of SECURITY=(1,x) or (2,x) in the PowerExchange DBMOVER configuration file, you must provide a valid user ID and password. For a data source or target on i5/OS or z/OS, you can enter a valid PowerExchange passphrase instead of a password. An i5/OS passphrase can be from 9 to 31 characters in length. A z/OS passphrase can be from 9 to 79 characters in length. A passphrase can contain uppercase and lowercase letters, numbers, spaces, and the following special characters:

' - ; # \ , . / ! % & * () _ + { } : @ | < > ?

Note: The first character is an apostrophe.

3. Click **Connect**.
4. Select the table or tables that you want to import. To select multiple tables, use the Shift key or Ctrl key or click **Select All**.
5. Click **OK**.
The source or target definition appears.
6. Click **Repository > Save**.

Working with Source Definitions for PowerExchange Change or Real-time (ODBC)

With PowerExchange ODBC, you can create source definitions to access change data for VSAM data sets, including CICS/VSAM, and ADABAS, Datacom, DB2, IDMS, IMS, MSSQL Server, and Oracle databases.

Importing a Change Data Capture Source Definition for PowerExchange ODBC

You can import PowerExchange extraction maps to create source definitions for CDC sources for which you use PowerExchange ODBC connections.

When you create capture registrations in the PowerExchange Navigator, corresponding extraction maps are automatically generated. You can also manually create them.

1. In the Source Analyzer, click **Sources > Import from Database**.
The **Import Tables** dialog box appears.
2. In the **Owner name** field, you can enter filter criteria to narrow the list of objects that are retrieved.

In PowerExchange extraction maps, the owner is the first qualifier of the extraction map name. The extraction map name has the following format:

DNinstance.regname_TABLENAME

Where:

- *D* is the default entry starts with D (any user-modified maps start with U)
- *N* is the database-specific identifier, such as **1** for DB2 for z/OS, **2** for IMS, **3** for DB2 for i5/OS, and so on.
- *instance* is the instance name chosen for the source registration
- *regname* is the registration name chosen for the source registration.

For example, a DB2 for z/OS extraction map name could be `dldsn9.testdb2_KJM723TB`.

When you import extraction maps to create CDC source definitions, the ODBC data source must have a DB Type of CAPX or CAPXRT. These DB Type values instruct the PowerExchange Listener to select extraction maps rather than data maps.

If the ODBC data source points to a PowerExchange Listener that is running with a PowerExchange security setting of SECURITY=(1,x) or (2,x) in the DBMOVER configuration file, you must provide a valid user name and password. For a data source on i5/OS or z/OS, you can enter a valid PowerExchange passphrase instead of a password. An i5/OS passphrase can be from 9 to 31 characters in length. A

z/OS passphrase can be from 9 to 79 characters in length. A passphrase can contain uppercase and lowercase letters, numbers, spaces, and the following special characters:

' - ; # \ , . / ! % & * () _ + { } : @ | < > ?

Note: The first character is an apostrophe.

3. Click **Connect**.
4. Select the table or tables that you want to import.
To select multiple tables, use the Shift key or Ctrl key or click **Select All**.
5. Click **OK**.
The source definition appears.
6. Click **Repository > Save**.

CHAPTER 10

Configuring Connections for PowerExchange ODBC

This chapter includes the following topics:

- [Configuring Connections for PowerExchange ODBC Overview, 195](#)
- [Working with Connections for PowerExchange ODBC, 195](#)

Configuring Connections for PowerExchange ODBC Overview

Before PowerCenter can access a source or target in a session, you must configure connections in the Workflow Manager. When you create or modify a session that reads from or writes to a database, you can select only configured source and target databases. Connections are saved in the repository. For PowerExchange ODBC, you configure relational database connections.

For more information about PowerExchange ODBC, see the *PowerExchange Reference Manual*.

Working with Connections for PowerExchange ODBC

This section describes the connection types used for extracting and loading data as well as how to configure connection attributes.

Connection Types for Extracting Source Data for PowerExchange ODBC

The type of connection you configure depends on the extraction mode but not on the source type. However, the source type does affect which ODBC data source is used in the Connect String when multiple Listeners and platform-specific parameters exist.

The following table describes the connection type to create based on extraction mode:

Extraction Mode	Application Connection or Relational Database Connection	Connection Type
Batch mode (nonrelational data map)	Relational Connection	ODBC with NRDB and NRDB2 Data Source
Batch mode (relational)	Relational Connection	ODBC with Data Source of appropriate database type (DB2, DB2400C, ADAUNLD, and so on.)
Change mode	Relational Database Connection	ODBC with CAPX Data Source
Real-time mode	Relational Database Connection	ODBC with CAPXRT Data Source

Connection Types for Loading Target Data for PowerExchange ODBC

All target connections are Relational Database Connection when loading data to PowerExchange targets.

The following table describes the target database types and their capability:

Database Type (Access Method)	Insert	Update	Delete
Adabas	Yes	Yes	Yes
Datacom	No	No	No
DB2 (DB2)	Yes	Yes	Yes
DB2 (DB2400C)	No	No	No
IDMS	No	No	No
IMS	Yes	Yes	Yes
Sequential (NRDB/NRDB2)	Yes	No	No
VSAM- KSDS (NRDB/NRDB2)	Yes	Yes	Yes
VSAM-ESDS (NRDB/NRDB2)	Yes	No	No
VSAM-RRDS (NRDB/NRDB2)	Yes	No	No

Configuring Attributes for Connections for PowerExchange ODBC

When you use ODBC connections, the majority of the attributes are specified in the ODBC data source or target rather than in the PowerCenter connection.

In Workflow Manager, source and target ODBC connections are defined as relational connections with a subtype of ODBC. For data sources, the ODBC connection definition process is the same for Batch, Change, and Real-time processing. The only difference in these connections is the ODBC data source.

The following table shows the connection attributes in the **Connection Object Definition** dialog box for an ODBC relational connection:

Connection Attribute	Required or Optional	Description
Name	Required	Name for the relational database connection.
User Name	Required	User name for access to the data source or target.
Password	Required	<p>A password for the specified user.</p> <p>To connect to a source or target on i5/OS or z/OS, you can enter a valid PowerExchange passphrase instead of a password. An i5/OS passphrase can be from 9 to 31 characters in length. A z/OS passphrase can be from 9 to 79 characters in length when you use ODBC. A passphrase can contain the following characters:</p> <ul style="list-style-type: none"> - Uppercase and lowercase letters - The numbers 0 to 9 - Spaces - The following special characters: ' - ; # \ , . / ! % & * () _ + { } : @ < > ? <p>Note: The first character is an apostrophe.</p> <p>Passphrases cannot include single quotation marks ('), double quotation marks ("), or currency symbols.</p> <p>On z/OS, the allowable characters in the IBM IRRPHREX exit do not affect the allowable characters in PowerExchange passphrases.</p> <p>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.</p> <p>To use passphrases, ensure that the PowerExchange Listener runs with a security setting of SECURITY=(1,N) or higher in the DBMOVER member. For more information, see "SECURITY Statement" in the <i>PowerExchange Reference Manual</i>.</p>
Connect String	Required	Name of the ODBC data source or target.
Code Page	Required	Code page for the PowerCenter Integration Service to use to extract the data from the data source.
Connection Environment SQL	Optional	Executes an SQL command with each database connection. Default is disabled.

CHAPTER 11

Working with Sessions for PowerExchange ODBC

This chapter includes the following topics:

- [Working with Sessions for PowerExchange ODBC Overview, 198](#)
- [Extracting Data from PowerExchange in Batch Mode \(ODBC\), 199](#)
- [Extracting Data from PowerExchange in Change and Real-time Mode \(ODBC\), 200](#)
- [Configuring a Session to Load Data to PowerExchange Targets \(ODBC\), 201](#)

Working with Sessions for PowerExchange ODBC Overview

After you create mappings in the PowerCenter Designer, you can create a session and use the session in a workflow to extract, transform, and load data. You create sessions and workflows in the Workflow Manager.

You can create a session in a workflow to extract data in batch, change, or real-time mode. You determine how you want the PowerCenter Integration Service to extract the data when you configure the session. You can also create a session to load data to a target.

After you configure the workflow, you must schedule the workflow.

Pipeline Partitioning for PowerExchange ODBC

Depending on your source or target database, you can increase the number of partitions in a pipeline to improve session performance. Increasing the number of partitions allows the PowerCenter Integration Service to create multiple connections to sources and targets and process partitions of data concurrently. While processing data, the PowerCenter Integration Service may process data out of sequence due to the varying rates at which the partitions process data.

When you create a session in a workflow, the Workflow Manager validates each pipeline in the mapping for partitioning. You can specify multiple partitions in a pipeline if the PowerCenter Integration Service can maintain data consistency when it processes the partitioned data.

Partition Types for Extracting Source Data for PowerExchange ODBC

If you are configuring a session to read source data, you can use partitioning if the session mode is batch.

The following table describes the partition types for partitioning points when extracting source data in batch mode:

Source Type	Partitioning Point	Partition Type
Nonrelational	Application Multi-Group Source Qualifier	Pass-through
Relational	Source Qualifier	Key range Pass-through

Partition Types for Loading Target Data for PowerExchange ODBC

If you are configuring a session to write data to DB2 for z/OS and DB2 for i5/OS targets, you can specify all partition types.

Extracting Data from PowerExchange in Batch Mode (ODBC)

To extract data in batch mode, you must select the appropriate ODBC relational connection and configure session properties. The connection you select and the session properties you configure differ depending on the type of source data.

Configuring Properties for Nonrelational Batch Mode Sessions (ODBC)

The relational connection for nonrelational sources should point to an ODBC data source that has a DBType of either NRDB or NRDB2. You should use a connection with the same DBType as used when the source was imported. That is, if you imported the source mapping by using NRDB, then use an NRDB ODBC connection in the session connection as opposed to an NRDB2.

Note: The only difference between NRDB and NRDB2 is whether a three or two tier naming conventions is used in the SQL statements to extract or load data. Nonrelational sources and targets mapped in PowerExchange can be referred to as either NRDB or NRDB2. The format is as follows:

- NRDB: schema.datamapname.table
- NRDB2: schema.datamapname_table

To configure properties for a nonrelational batch mode session:

1. In the Task Developer, double-click a session with a nonrelational source to open the session properties.
2. Click the Sources view on the Mapping tab.
3. In the Reader field of the Readers settings, verify that Relational Reader is selected.
4. In the Connections Value field, select a nonrelational ODBC connection, that is, NRDB or NRDB2.
5. In the Properties settings, configure the Owner Name attribute.

At minimum, the schema name of the PowerExchange data map, or the Owner name that is displayed when the source mapping is edited, must be specified to correctly construct the SQL statement during

execution. PowerExchange SQL Escape Sequences can also be specified in this attribute field to override specifications in the data map.

For a list of the available SQL escape sequences, see the *PowerExchange Reference Manual*.

The following example indicates how the Owner Name attribute can be configured to provide the schema name for the source (seq) as well as an override for the physical file name in the data map (dtldsn=new.dataset.name):

```
seq{dtldsn=new.dataset.name}
```

Note: SQL escape sequences and the Owner Name can be specified in any order in the attribute field.

6. Click OK.

Configuring Properties for Relational Batch Mode Sessions (ODBC)

To extract data from a relational source in batch mode, you must select the appropriate ODBC relational connection and configure session properties. The relational connection should point to an ODBC data source that has the DBType of the source (DB2, DB2400C, DB2UDB, MSSQL or Oracle). You can then configure the session properties as you would any other PowerExchange ODBC source.

To configure a properties for a relational batch mode session:

1. In the Task Developer, double-click a session with a relational source to open the session properties.
Click the **Sources** view on the **Mapping** tab.
2. In the Reader field of the Readers settings, verify that Relational Reader is selected.
3. In the **Connections Value** field, select the appropriate relational ODBC connection.
4. In the **Properties** settings, configure the **Owner Name** attribute.
At minimum, the owner name of the source table must be specified to correctly construct the SQL statement during execution. However, if the owner name was specified in the source mapping, do not specify it here.
5. Click **OK**.

Extracting Data from PowerExchange in Change and Real-time Mode (ODBC)

To extract data in change and real-time mode, you must select an ODBC relational connection and configure session properties. The connection you select for change mode should refer to an ODBC data source with a DBType of CAPX whereas a real-time mode connection should refer to an ODBC data source with a DBType of CAPXRT.

When you use ODBC to extract data from PowerExchange in either Change or Real-time mode, restart is handled from within PowerExchange.

Configuring Properties for Change and Real-time Mode Sessions (ODBC)

Use the following procedure to configure a session for Change or Real-time mode:

1. In the Task Developer, double-click a session with a relational source to open the session properties.

2. Click the Sources view on the Mapping tab.
3. In the Reader field of the Readers settings, verify that Relational Reader is selected.
4. In the Connections Value field, select a connection that points to an ODBC data source with the DBType of CAPX (for Change) or CAPXRT (for Real-Time) and the appropriate Location value.
5. In the Properties settings, configure the Owner Name attribute.

At minimum, the schema name of the source extraction map must be specified to correctly construct the SQL statement during execution. This name is the first qualifier of extraction map name shown in the PowerExchange Navigator. It can also be determined by checking the Owner Name in the source mapping. PowerExchange SQL Escape Sequences can also be specified in this attribute field to override specifications in the data map. For a complete list of the SQL Escape Sequences available, see the *PowerExchange Reference Manual*.

The following example indicates the manner in which the Owner Name attribute can be configured to provide the schema name for the source (d6vsam) as well as an override for the application name specified in the DBQual2/Application Name field in the ODBC data source (dtlapp=new_appname):

```
{dtlapp=new_appname}d6vsam
```

Note: SQL escape sequences and the Owner Name can be specified in any order in the attribute field.

6. Click OK.

Configuring a Session to Load Data to PowerExchange Targets (ODBC)

To load data to a PowerExchange target, you must select a relational database connection. Then configure properties for the session as you would any other relational target.

To configure a session to load data to a PowerExchange target:

1. In the Task Developer, double-click a session with a relational source to open the session properties.
2. Click the Targets view on the Mapping tab.
3. In the Writers settings, ensure that Relational Writer is selected.
4. In the Connections Value field, select a connection which points to an ODBC data source with appropriate DBType and Location value.
5. In the Properties settings, configure the Table Name Prefix attribute.

At minimum, the schema name of the target table (if relational) or the PowerExchange data map (if nonrelational) must be specified in order to correct construct the SQL statement during execution.

6. Click OK.

CHAPTER 12

PowerExchange Restart and Recovery

This chapter includes the following topics:

- [PowerExchange Restart and Recovery Overview, 202](#)
- [Restart and Recovery with PowerExchange ODBC, 202](#)

PowerExchange Restart and Recovery Overview

Recovery and restart need to be considered when designing and configuring sessions and workflows by using either PowerExchange Change or Real-Time. The considerations differ depending upon whether the session uses PowerExchange Client for PowerCenter (PWXPC) or PowerExchange ODBC. This is primarily due to the differences in where the restart information is maintained.

With PowerExchange ODBC, the restart information is controlled and maintained on the PowerExchange Listener platform in the internal change information file (CDEP) by using the application name specified in the connection.

Restart and Recovery with PowerExchange ODBC

PowerExchange ODBC uses the restart tokens associated with application names in the CDEP file to determine the starting extraction point for a session. The CDEP file used is the one maintained by PowerExchange on the platform targeted by the extraction. The CDEP file contains both begin and end restart tokens for each extraction. This differs from PWXPC where only the end restart tokens are maintained in the restart token file.

The restart tokens determine the baseline point from which to extract the change data for the source database. Since both start and end restart tokens are available when you use ODBC, the extraction type decides which ones will be used for restart. The extraction type is specified in the ODBC connection in the CAPXTYPE parameter. This can be overridden at the session level by using the SQL Escape Sequence DTLXTYPE.

If the extraction type specified is Since Last (SL) then the extraction will commence at the last successful ending point. If it is Restart (RS) then it will commence at the starting point of either the last successful extraction that was run or the one that is specified with the Reset Start Point in the extraction application dialog box in the PowerExchange Navigator.

Note: Unique application names should be used for each for each session to prevent any conflict in the CDEP between multiple sessions.

The CDEP information for an extraction is only updated when a session ends successfully. Failed sessions do not update the restart token information in the CDEP with the progress so far.

Warning: Failed sessions that are restarted will extract data from the last successful session (based on CAPXTYPE specified). This means that there is the possibility that duplicated changes will be sent to the target. In the event of a failure, you must either restore the targets to match the restart point prior to restarting the session or design your PowerCenter session to handle the possibility of duplicate records.

Managing Restart Tokens with PowerExchange ODBC

PowerExchange on the source platform reads the restart tokens from the CDEP for each source included in the session. The first time a session is run with a new application name, an entry for it will be automatically created in the CDEP if it does not already exist.

If the application name already exists in the CDEP, PowerExchange will use the appropriate restart tokens to commence extraction from the change stream. Which restart tokens are used depends upon the extraction type (SL or RS) and whether new tokens have been supplied by DTLUAPPL since the last run. If new tokens have been supplied by DTLUAPPL, then those tokens will be used regardless of the extraction type.

If there are sources included that have newer restart points, they will not be provided any records from the change stream until their restart points have been reached. This prevents sources from being passed records that they processed in the previous extraction run.

After a session completes successfully, PowerExchange updates the CDEP with the ending restart tokens.

CDEP Restart Tokens Example

The following example shows the contents of restart tokens in the CDEP for an application that contains a restart token file for the DB2 sources DB2DEMO1, DB2DEMO2, and DB2DEMO3:

```
Application name=<odbc_db2demo13ac> Rsttkn=<3> Ainseq=<0> Preconfig=< >
FirstTkn   =<D2D1D4D340400000003ED5D600000000>
LastTkn    =<D2D1D4D340400000003ED5D600000000>
CurrentTkn=<>
Registration name=<db2demo1.1> tag=<DB2DSN7db2demo11>
Sequence=<000007337D8A0000000000000007337D8A00000000>
Restart   =<D2D1D4D340400000007337D0200000000>
Registration name=<db2demo2.1> tag=<DB2DSN7db2demo21>
Sequence=<000007337D8A0000000000000007337D8A00000000>
Restart   =<D2D1D4D340400000007337D0200000000>
Registration name=<db2demo3.1> tag=<DB2DSN7db2demo31>
Sequence=<000007337D8A0000000000000007337D8A00000000>
Restart   =<D2D1D4D340400000007337D0200000000>
```

The application name `odbc_db2demo13ac` was created by using DTLUAPPL with RSTTKN GENERATE. The following example shows the control cards:

```
mod APPL odbc_db2demo13ac DSN7 RSTTKN GENERATE
add rsttkn db2demo1
add rsttkn db2demo2
add rsttkn db2demo3
end APPL odbc_db2demo13ac
print appl odbc_db2demo13ac
```

You can use either ADD APPL or MOD APPL for a new application name as with MOD APPL DTLUAPPL will create the application name if it does not already exist.

After the new restart tokens are provided through DTLUAPPL, the session using this application name can be started and it will use this restart tokens.

APPENDIX A

PowerExchange Interfaces for PowerCenter Tips

This appendix includes the following topics:

- [Organizing Sources by Map Type in the Designer, 204](#)
- [Filtering Source Data with PWXPC, 205](#)
- [DTLREXE to Submit MVS Batch Jobs, 207](#)
- [Using Pre-SQL and Post-SQL Statements to Manage Files and Tables, 208](#)

Organizing Sources by Map Type in the Designer

Source metadata imported in Source Analyzer in the PowerCenter Designer with PWXPC is stored into sub-folders based on source type and location. The user cannot choose the sub-folder name and the table names within those sub-folders must be unique.

With the PWXPC Import from PowerExchange dialog box, these folder names have the following format: `<source_type_location>`. The *source_type* value equates to the Source Type specified in the Import from PowerExchange dialog box. The *location* value equates to the Location specified in the Import from PowerExchange dialog box and is a user-specified name in the PowerExchange dbmover.cfg file.

For example, if the Location is MVS2 and the Source Type is IMS, then the folder will be called IMS_MVS2. All IMS sources imported from that location will be stored into that folder. You can use the name you assign for the location as a method of organizing your source metadata in Designer.

There are some nonrelational source types where the schema name is different but map name can be the same for both PowerExchange data map and extraction map (CDC Datamap) created from that data map.

To illustrate, assume that the following information is used to create the PowerExchange data map for an IMS database:

- IMS database name is **IMS1T01**. This name is used as the PowerExchange data map name.
- A segment in that database results in a table in the data map that is called **IMSSEG1**.
- The PowerExchange schema name that used when creating the data map is **IMS**.

The PowerExchange map name (in the NRDB2 form) is constructed as follows:

schema_name.datamap_name_table_name

In this example, the PowerExchange data map name is IMS.IMS1T01_IMSSEG1. If you use the IMS database name for the Capture Registration Name as well, the resulting extraction map name will be d2reconid.IMS1T01_IMSSEG1.

The map name for both the data map and extraction map are the same. Since the schema name is not used in the source definition table name, the names will be exactly the same when imported in Designer. As a result, only one will be able to be imported from the same Location. So, there are two choices:

1. Use a different name for the capture registration so that a unique extraction map name is created.
2. Use a different Location name to import data maps and extraction maps.

If you want to use the same names for both the data map name and the capture registration name, then you will need to use a different location name in the Import from PowerExchange to place these two source definitions in different folders.

To help organize sources in Designer, use location names in the PowerExchange dbmover.cfg that indicate the type of data you are retrieving. Creating multiple NODE= statements (location names) for the same listener is perfectly acceptable. In this example, if you create a NODE called CDCMAPS in the dbmover.cfg and use this as the Location when importing the IMS extraction map, it will be stored in a sub- folder called IMS_CDCMAPS.

This strategy of separating extraction maps from regular relational and nonrelational source metadata can be a useful way of organizing source metadata in Designer.

Filtering Source Data with PWXPC

You can use PWXPC to filter data at the source so PowerExchange only sends the filtered rows to the PowerCenter session. There are many reasons why you might want to do this. Some examples of scenarios in which this capability is useful are:

- For some source types, such as VSAM and IMS, you cannot limit change capture changes based on only certain columns changing.
- You can, for certain source types such as DB2 for i5/OS, create capture registrations in PowerExchange that register only the columns you select. However, if the RDBMS logs are used directly for extraction and do not support selective column capture, PowerExchange extracts the row even if none of the columns of interest have changed.
- The source type is one where either PowerExchange itself or the RDBMS will only capture changes based on columns of interest. However, additional or all columns are registered for capture because other extractions require them.
- You only want to extract columns with a specific value. For example, you want to read all of the columns in a table for a specific customer.

You can use the **Filter Override** attribute in the Session Properties to filter records for sources that use PWX Batch, CDC Change, and CDC Real Time connections. When you specify filter conditions in the **Filter Override** attribute, PWXPC includes a WHERE clause with the filter conditions on the SELECT statement that is passed to PowerExchange. You must use proper SQL syntax for these overrides to prevent SQL failures. Additionally, PowerExchange supports a limited set of SQL syntax for nonrelational sources, which includes bulk and change data.

For more information about NRDB SQL syntax that PowerExchange supports, see the *PowerExchange Reference Manual*.

In the filter condition, you can specify any column that exists in the source definition, including PowerExchange-generated columns such as DTL__CAPX columns, change indicator columns (DTL__CI) and before image columns (DTL__BI).

PWXPC supports two forms of filter condition syntax. For single record source definitions such as CDC extraction maps, relational tables, or single record nonrelational data maps, specify a single filter condition statement. You can specify a single filter condition or join numerous filter conditions by using the conditional operands that PowerExchange supports for NRDB SQL statements. For example:

```
column1 is NULL and column2='A'
```

For multi-record nonrelational source definitions, you can also use the following syntax:

```
group_name1=filter_condition;group_name2=filter_condition;...
```

Use the *group_name* form to specify filter conditions for one or more record types in a multi-record source definition. To apply a filter condition to all records in a multi-record source definition, use a single filter condition without *group_name*. You cannot combine single filter conditions with *group_name* conditions.

Filter Override Examples

The following examples show how to use filter overrides for CDC and multi-record source definitions.

Example 1: Filter Change Data with Change Indicator Columns

To filter the change data for this source table to include changes when the ACCOUNT column was changed, code the following filter condition:

```
DTL__CI_ACCOUNT='Y'
```

To filter change data to include changes where the ACCOUNT column was changed to 9999, code the following filter condition:

```
DTL__CI_ACCOUNT='Y' and ACCOUNT=9999
```

Note: To use DTL__CI change indicator or DTL__BI before-image columns in filter conditions, you must alter the extraction map in PowerExchange Navigator to define these columns.

Example 2: Filter Records in a Multi-Record VSAM Source Definition

In the following example, a multi-record VSAM source definition contains the following four records:

- V07A_RECORD_LAYOUT
- V07B_RECORD_LAYOUT
- V07C_RECORD_LAYOUT
- V07D_RECORD_LAYOUT.

Each record has unique field names, which means you must use the *group_name* syntax for filter conditions. To filter data for the first two records, the filter condition uses the following *group_name* syntax:

```
V07A_RECORD_LAYOUT=V07A_RECORD_KEY=1;V07B_RECORD_LAYOUT=V07B_RECORD_KEY=2
```

No filtering is done on the other two records in the multi-record source definition.

PWXPC creates a SELECT statement for each record in the multi-record source definition. Because the filter override uses the *group_name* syntax and only specifies two records, PWXPC includes the WHERE clause on the SELECT statements for records V07A_RECORD_LAYOUT and V07B_RECORD_LAYOUT.

To use a single filter condition to filter all records in a multi-record source definition, the filter condition must apply to all records. Otherwise, the session fails. For example, if all records in a multi-record source definition

contain a column called ADDRESS, you can code the following filter condition to select all records where ADDRESS is not null:

```
ADDRESS is not NULL
```

PWXPC applies this filter condition to all records in the multi-record source definition by including a WHERE clause with the condition on the SELECT statement for each record.

DTLREXE to Submit MVS Batch Jobs

PowerExchange provides a utility program called DTLREXE which can be used to cause remote execution of a program on another platform running PowerExchange. One option of this utility is the SUBMIT option which provides the ability to submit a MVS batch job remotely.

This utility can be invoked in PowerCenter workflows with PowerCenter Command Tasks. Command tasks can either be stand-alone tasks or pre/post session commands. Which you choose depends upon what you want to do. There are advantages in using a stand-alone command task for repeatable functions as it only needs to be coded once and can then be used by numerous workflows. On the other hand, it is possible to easily configure a session to fail if a pre-session command task fails.

Some examples of the type of tasks that can be performed with DTLREXE PROG=SUBMIT are:

- Truncate a database table prior to loading data into it in a session. This is useful for database types for which PowerCenter does not support truncate, such as Adabas.
- Notify a MVS-based job scheduler that the workflow is starting or ending. Some job scheduler products provide batch posting utilities and they can be utilized with DTLREXE to submit a batch job.
- Unload a database to a flat file so it can then be used in a session to load another database.
- Clean up DB2 bulk load files when the session completes successfully.
- Submit any type of MBS Batch JOB for which waiting for the completion and returning a set of messages is required.

In the **Edit - Command** dialog box, you can enter a command. The following code is an example of a DTLREXE PROG=SUBMIT command entered as a pre-session command:

```
dtlrexe prog=submit loc=mvs fn=\"dtlusr.jcl(db2load)\"  
mode=(job,wait) output=dtlusr.output  
result=c:\submit\output\output.txt  
uid=user01 pwd=pass01
```

In this example, the DTLREXE command specifies mode=(job,wait) which means that the DTLREXE will wait for the job to complete. This, in turn, will cause the session to wait until this pre-session command completes. In the “Error Handling” section of the Config Object, you can specify how to handle errors for pre-session commands in the **On Pre-session command task error** field.

Note: Ensure that the JOB submitted through DTLREXE includes the appropriate DTLNTS steps if WAIT mode is requested. The PowerExchange RUNLIB, in member DTLREXE, contains sample JCL to be used with DTLREXE that includes the required DTLNTS steps.

If you use a stand-alone command task to submit a batch JOB with DTLREXE, there are no session configuration options to check for success or failure. If you want to test the status of the command task in the following session, you will need to use one of the task-specific workflow variables available in the Workflow Manager, that is, PrevTaskStatus or Status. These variables can be used in link conditions to test the status of tasks in a workflow.

The link condition is created by double-clicking on the link between the DTLREXE command task and the s_bulk_db2demo123_db2demoabc session to which it is connected. This action invokes the Expression Editor, enabling you to add the test to ensure that the DTLREXE command task succeeded.

Using Pre-SQL and Post-SQL Statements to Manage Files and Tables

You can define SQL statements in the **Pre SQL** and **Post SQL** session properties for a nonrelational target to manage files and tables.

You can:

- Create an empty file
- Truncate a target table
- Insert rows into a target table

Creating Empty Files

PowerExchange can automatically create empty sequential files or new generations of a Generation Data Group (GDG).

To create an empty sequential file or generation of a GDG, code a CREATEFILE command in the **Pre SQL** or the **Post SQL** session property for any PowerExchange source or target. PowerCenter uses the connection information from the source or target on which you code the command and passes the CREATEFILE command to PowerExchange. PowerExchange processes the command and creates the file.

To create a file before the session runs, code the CREATEFILE command in the **Pre SQL** property. To create a file after the session runs, code the CREATEFILE command in the **Post SQL** property.

Note: A PowerCenter session with a PowerExchange target can create a new flat file or sequential data set even if you do not code the CREATEFILE command. When the session writes the first record to a file that does not exist, PowerExchange creates the file, using any data set allocation attributes that you specify in the session properties, if applicable. However, if the session does not write any records to the file, PowerExchange does not create it. In this case, if a task later in the workflow requires that the file exist, the task fails.

The following rules and guidelines apply to the CREATEFILE command:

- You can code CREATEFILE commands for any source or target that uses a PowerExchange connection for which the **Pre SQL** or **Post SQL** property is available.
- You do not need to create a PowerExchange data map or PowerCenter source or target definition for the file name that you code in the command.
- For z/OS and i5/OS, you can specify allocation information for the file by coding allocation parameters on the CREATEFILE command. Otherwise, the PowerExchange Listener that allocates the files uses allocation values from the DBMOVER configuration file, or default values.

Note: Because PowerExchange creates files either before or after a session runs, it does not use any allocation information specified in session properties for nonrelational sources or targets.

- In certain cases, you must specify the **Pre SQL run once per Connection** attribute along with the **Pre SQL** attribute. Select the **Pre SQL run once per Connection** attribute in either of the following cases:
 - You specify a CREATEFILE statement in the **Pre SQL** attribute for a session that uses writer partitioning. If you do not select **Pre SQL run once per Connection**, the session tries to run the statement once for each partition.
 - You specify a CREATEFILE statement that creates a new generation of a GDG or creates an empty file in the **Pre SQL** attribute for a session that performs a multiple-record write. If you do not select **Pre SQL run once per Connection**, the session creates a generation or tries to create a new empty file for each record that the session writes.

Creating Empty Files in a PowerCenter Session

You can create an empty flat file, a new generation of a GDG, or a sequential data set in a PowerCenter session.

1. In Workflow Manager, right-click the appropriate task in the Task Developer or the workflow in Workflow Designer.
2. Click **Tasks > Edit**.
The **Edit Tasks** dialog box appears.
3. Click the **Mapping** tab and select the source or target.
4. In the **Pre SQL** or **Post SQL** attribute under **Session-Level Properties**, enter the CREATEFILE command. Use the following syntax:

```
<CMD>CREATEFILE FN=file_name parameter2 parameter3 ...
```

The following table describes the parameters that you can enter for the command:

Parameter	Operating System	Description
BS	z/OS	Block size. Valid values are 1 to 32760. If RECFM=VB, PowerExchange might add up to 4 bytes to the value that you specify to determine the actual block size. For more information, see "Block Size and Record Size in Variable-Blocked Data Sets" on page 211 .
CLOSEDEALLOC	z/OS	Deallocate the data set at CLOSE rather than at step termination (FREE=CLOSE). Valid value is Y.
EPWD	z/OS or i5/OS	An encrypted password for the specified user ID in the UID parameter. Alternatively if you are creating a file on z/OS, you can enter an encrypted PowerExchange passphrase. Do not encrypt a passphrase that contains invalid characters, such single quotation marks ('), double quotation marks ("), or currency symbols. Do not also specify the PWD parameter.

Parameter	Operating System	Description
FN	All	<p>Required. File or data set name to be created. You must enclose file names that include parentheses in double quotes (").</p> <p>To create a new generation of a GDG, code: <code>FN="gdg_base_name(+1)"</code></p> <p>Restriction: If you create a new generation, the GDG base name must exist. The PowerExchange Listener on z/OS that creates the new generation must specify Y for the GDGLOCATE parameter in the DBMOVER configuration file on z/OS.</p>
LRECL	z/OS or i5/OS	<p>Logical record length.</p> <p>Valid values are 1 through 32756.</p> <p>If RECFM=VB, specify the maximum number of data bytes in a logical record. PowerExchange adds 4 to this value for the record descriptor word (RDW). For more information, see "Block Size and Record Size in Variable-Blocked Data Sets" on page 211.</p>
PWD	z/OS or i5/OS	<p>A password for the specified user ID in the UID parameter.</p> <p>If you are creating a file on i5/OS or z/OS, you can enter a valid PowerExchange passphrase instead of a password. An i5/OS passphrase can be from 9 to 31 characters in length. A z/OS passphrase can up to 128 characters in length if you use a PWXPC connection or up to 79 characters in length if you use an ODBC connection. A passphrase can contain the following characters:</p> <ul style="list-style-type: none"> - Uppercase and lowercase letters - The numbers 0 to 9 - Spaces - The following special characters: ' - ; # \ , . / ! % & * () _ + { } : @ < > ? <p>Note: The first character is an apostrophe.</p> <p>Passphrases cannot include single quotation marks ('), double quotation marks ("), or currency symbols.</p> <p>Note: 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.</p> <p>Do not also specify the EPWD parameter.</p>
MODELDCB	z/OS	Model DCB to be used for the file creation. Generally, this is only required for GDG data sets that are not SMS-managed.
RECFM	z/OS or i5/OS	Record format.
RELEASE	z/OS	<p>Release unused, allocated space on CLOSE.</p> <p>Valid value is Y.</p>
RELPOS	z/OS	<p>Relative position of the data set on a tape volume, or data set sequence number.</p> <p>Valid values are 1 to 512.</p>

Parameter	Operating System	Description
SPACE	z/OS	Space allocation parameters. Use the following syntax: SPACE= (U, P, S) Where: <ul style="list-style-type: none"> - U is unit type. Valid values are: T for tracks or C for cylinders. - P is primary space. Valid values are 1 through 16777215 tracks or the equivalent in cylinders. - S is secondary space. Valid values are 1 through 16777215 tracks or the equivalent in cylinders. <p>If you enter a primary or secondary space value greater than 65535 tracks, coordinate with the space management team for the target system to verify that such a large allocation is supported.</p>
UID	z/OS or i5/OS	User ID. This parameter is required if the PowerExchange Listener is configured for security (SECURITY=1 or 2). Also specify either PWD or EPWD, but not both.
UNIT	z/OS	Unit type.
VOLSER	z/OS	Volume serial.

Use a space to separate parameters in the command.

5. Click **OK**.

Empty File - Example

The following example CREATEFILE statements show how to create a new generation of a GDG and a flat file.

To create a new generation with a record length of 80 bytes and a block size of 8880 in a GDG data set called MY.GDG, code the following command:

```
<CMD>CREATEFILE FN="MY.GDG(+1)" LRECL=80 BS=8880 RECFM=PS
```

The PowerExchange Listener on MVS that creates the new generation uses its DBMOVER member to determine the remaining DCB and space attributes for this file.

To create a new flat file called my.flat.file, code the following command:

```
<CMD>CREATEFILE fn=my.flat.file
```

Block Size and Record Size in Variable-Blocked Data Sets

When you specify RECFM=VB in the CREATEFILE command to create a variable-blocked data set, the block size and the record size of the data set vary slightly from the values that you specify for BS and LRECL.

When RECFM=VB, the value that you specify for LRECL represents the maximum number of data bytes in a record. To reserve space for the record descriptor word (RDW), PowerExchange adds four bytes to the LRECL value that you specify to determine the actual record length in the new data set.

Also, When RECFM=VB, PowerExchange might add up to four bytes to the value that you specify for BS. In this way, PowerExchange ensures that the actual block size value is at least four bytes greater than the actual record size value. These four bytes are reserved for the block descriptor word (BDW).

For example, suppose you include the following command in the **Pre SQL** attribute:

```
<CMD>CREATEFILE FN=JSMITH.RVLOAD.TEST UNIT=SYSDA RECFM=VB LRECL=27994 BS=27998 SPACE=(C,
50,50) RELEASE=Y CLOSEDEALLOC=Y
```

The resulting data set has the following attributes:

```
Record format . . . : VB
Record length . . . : 27998
Block size . . . . : 28002
```

In this example, PowerExchange increases the record size from 27994 to 27998. Also, to ensure that the block size is at least four bytes greater than the record size, PowerExchange increases the block size from 27998 to 28002.

Depending on the track size, the resulting block size might be greater than half a track, resulting in wasted space. To avoid this problem, ensure that the value that you specify for LRECL represents the number of data bytes in the record and does not include an additional four bytes for the RDW.

Truncating a Target Table

For nonrelational target types, PowerExchange and PowerCenter support clearing target tables before writing to them in a PowerCenter workflow.

You can clear a target table in the following ways:

- Include SQL to truncate the table in the **Pre SQL** session attribute for the target table.
- Select the **Truncate target option** session property for a VSAM target that uses a PWXPC connection.
- Select the **Truncate Target Table** property for a nonrelational target that uses an PowerExchange ODBC connection.

The first two tokens in the SQL can be either TRUNCATE TABLE or DELETE FROM. The final tokens in the SQL define the fully qualified table from which the data map can be determined. For example, you could use one of the following statements:

```
TRUNCATE TABLE schema.map_table
DELETE FROM schema.map_table
```

If you select the **Truncate target option** session property or the **Truncate Target Table** ODBC property, PowerExchange generates a statement, which you cannot modify, in one of these forms.

The system loads the data map for the table and determines the underlying map type.

For map types of ESDS, KSDS, RRDS, SEQ, or TAPE, the file defined in the data map (or overridden at run time) is opened in write mode and closed so that existing contents of the file are cleared. This method of clearing files is useful because otherwise, if no data is processed in a workflow, the output file might contain records from an earlier run that are retained. PowerExchange defers opening of output files until actual data is being written.

The entire file is always cleared. In the case of multiple-record files, any of the table names can be chosen and data for all tables is cleared. Although it is physically possible for an individual table to be cleared from KSDS and RRDS files, you cannot achieve this result by defining TRUNCATE TABLE or DELETE FROM in the **Pre SQL** session property. It might be possible to clear part of a KSDS file using DELETE SQL with a WHERE clause that is run as part of a workflow, but not by defining the statement in the **Pre SQL** session property.

Clearing a target table by using one of these methods is not supported for other data map types.

Note: VSAM data sets must be defined with the REUSE option for table truncation to function correctly.

Inserting Rows into a Table

A PowerCenter workflow can write to NRDB target types ADABAS, IMS, SEQ, TAPE, and VSAM. For these target types, it is possible to execute INSERT SQL to insert rows with hard-coded data values into the output table.

APPENDIX B

Datatypes and Code Pages

This appendix includes the following topics:

- [Datatypes and Code Pages Overview, 214](#)
- [PowerExchange Nonrelational Datatypes and Transformation Datatypes, 214](#)
- [Restrictions on Relational Datatypes, 216](#)
- [Reading and Writing Binary Data in PowerExchange Client for PowerCenter, 218](#)
- [Code Pages, 218](#)

Datatypes and Code Pages Overview

PowerCenter uses the following datatypes when reading source data, transforming the data, and writing target data:

- **Native datatypes.** Specific to the source and target databases or PowerExchange. Native datatypes appear in source and target definitions.
- **Transformation datatypes.** Generic datatypes that appear in transformations. The PowerCenter Integration Service uses the datatypes to move data across platforms.

PowerExchange Nonrelational Datatypes and Transformation Datatypes

You import PowerExchange data maps to create nonrelational source definitions. The datatypes of the fields in the source definition match the datatypes for the corresponding fields in the data map. These datatypes map to PowerCenter transformation datatypes. The transformation datatypes are displayed in the Application Multi-Group Source Qualifier for the nonrelational source and other transformations in a mapping.

Note: In PowerExchange, you can specify a precision of up to 31 for certain field datatypes, such as PACKED. These fields are mapped to the Decimal transformation datatype in PowerCenter. If you enable high precision in the session properties, PowerCenter supports a maximum precision of 28 for the Decimal datatype. If you do not enable high precision, or if you specify a precision greater than 28, PowerCenter uses the Double datatype with a precision of 16 and data rounding might occur.

The following table shows the PowerExchange nonrelational datatypes and the corresponding transformation datatypes:

PowerExchange Datatype	Precision	Transformation Datatype	Range
BIN	10	Binary	1 to 104,857,600 bytes You can pass binary data from a source to a target, but you cannot perform transformations on binary data. PowerCenter does not support binary data for COBOL or flat file sources.
CHAR	10	String	1 to 104,857,600 characters Fixed-length or varying-length string.
DATE	10	Date/Time	Jan 1, 0001 A.D. to Dec 31, 9999 A.D. Combined date/time value, with precision to the nanosecond.
DOUBLE	18	Double	Precision of 15 digits Double-precision floating-point numeric value.
FLOAT	7	Double	Precision of 15 digits Double-precision floating-point numeric value.
NUM8	3	Small Integer	Precision of 5 and scale of 0 Integer value.
NUM8U	3	Small Integer	Precision of 5 and scale of 0 Integer value.
NUM16	5	Small Integer	Precision of 5 and scale of 0 Integer value.
NUM16U	5	Integer	Precision of 10 and scale of 0 Integer value.
NUM32	10	Integer	Precision of 10 and scale of 0 Integer value.
NUM32U	10	Double	Precision of 15 digits Double-precision floating-point numeric value.
NUM64	19	Decimal	Precision 1 to 28 digits, scale 0 to 28 Decimal value with declared precision and scale. Scale must be less than or equal to precision. If you pass a value with negative scale or declared precision greater than 28, the PowerCenter Integration Service converts it to a double.
NUM64U	19	Decimal	Precision 1 to 28 digits, scale 0 to 28 Decimal value with declared precision and scale. Scale must be less than or equal to precision. If you pass a value with negative scale or declared precision greater than 28, the PowerCenter Integration Service converts it to a double.
NUMCHAR	-	String	1 to 104,857,600 characters Fixed-length or varying-length string.

PowerExchange Datatype	Precision	Transformation Datatype	Range
PACKED	15	Decimal	Precision 1 to 28 digits, scale 0 to 28 Decimal value with declared precision and scale. Scale must be less than or equal to precision. If you pass a value with negative scale or declared precision greater than 28, the PowerCenter Integration Service converts it to a double.
TIME	5	Date/Time	Jan 1, 0001 A.D. to Dec 31, 9999 A.D. Combined date/time value, with precision to the nanosecond.
TIMESTAMP	5	Date/Time	Jan 1, 0001 A.D. to Dec 31, 9999 A.D. Combined date/time value, with precision to the nanosecond.
UNPACKED	15	Decimal	Precision 1 to 28 digits, scale 0 to 28 Decimal value with declared precision and scale. Scale must be less than or equal to precision. If you pass a value with negative scale or declared precision greater than 28, the PowerCenter Integration Service converts it to a double.
UZONED	15	Decimal	Precision 1 to 28 digits, scale 0 to 28 Decimal value with declared precision and scale. Scale must be less than or equal to precision. If you pass a value with negative scale or declared precision greater than 28, the PowerCenter Integration Service converts it to a double.
VARBIN	10	Binary	1 to 104,857,600 bytes You can pass binary data from a source to a target, but you cannot perform transformations on binary data. PowerCenter does not support binary data for COBOL or flat file sources.
VARCHAR	10	String	1 to 104,857,600 characters Fixed-length or varying-length string.
ZONED	15	Decimal	Precision 1 to 28 digits, scale 0 to 28 Decimal value with declared precision and scale. Scale must be less than or equal to precision. If you pass a value with negative scale or declared precision greater than 28, the PowerCenter Integration Service converts it to a double.

Restrictions on Relational Datatypes

With certain exceptions, the PowerExchange Client for PowerCenter (PWXPC) supports the same datatypes for DB2 for z/OS, DB2 for i5/OS, and DB2 for Linux, UNIX, and Windows that PowerCenter supports. PWXPC also supports the same Oracle and Microsoft SQL Server datatypes that PowerCenter supports.

LOB Datatypes

PowerExchange supports LOB datatypes in DB2 for i5/OS and DB2 for z/OS source tables in bulk data movement sessions. LOB datatypes include BLOB, CLOB, and DBCLOB.

The following restrictions apply to DB2 tables that includes LOB columns:

- You cannot use an unload file as a data source.
- You can use an image copy as a data source only if the LOB columns are inline.
- You cannot perform an insert into a target table that contains LOB columns.

For relational data sources other than DB2 for i5/OS and DB2 for z/OS, PowerExchange does not support LOB datatypes. PWXPC does not include LOB columns when you import metadata for tables that include these datatypes.

DB2 for z/OS TIMESTAMP Datatype

How PowerCenter handles DB2 for z/OS TIMESTAMP columns depends on the DB2 version.

For DB2 versions earlier than version 10, the DB2 TIMESTAMP datatype maps to the PowerCenter transformation date/time datatype. The date/time datatype treats subsecond values as 32-bit integer values of up to nine digits.

The following table summarizes how the source TIMESTAMP datatype maps to the transformation date/time datatype for DB2 versions earlier than version 10:

DB2 Datatype	Range	Transformation Datatype	Range
TIMESTAMP	26 bytes Precision 26, scale 6 Precision is to the microsecond.	date/time	Jan 1, 0001 A.D. to Dec 31, 9999 A.D. Precision is to the nanosecond.

DB2 10 for z/OS introduced support for extended-precision TIMESTAMP columns, which can have subsecond values of up to 12 digits. The PowerExchange Client for PowerCenter supports DB2 10 extended-precision timestamps. Because the transformation date/time datatype does not support this level of extended precision, the DB2 10 TIMESTAMP datatype maps to the transformation string datatype, except when scale is 6. To maintain compatibility with pre-version 10 TIMESTAMP columns, which always have a scale of 6, all DB2 TIMESTAMP columns that have a scale of 6 map to the date/time datatype.

The following table summarizes how the DB2 10 extended-precision TIMESTAMP datatype maps to transformation datatypes based on scale:

Scale	Precision	Transformation Datatype
6	26	date/time
0	19	string
1 to 5 or 7 to 12	20+scale	string

When you write DB2 TIMESTAMP data to a DB2 target, the source and target must have the same scale. Otherwise, unpredictable results can occur. If the DB2 source TIMESTAMP field has a scale greater than 9 and the corresponding DB2 target TIMESTAMP field has a scale of 6, the value is truncated before it is written to the target.

When you write DB2 10 TIMESTAMP data to a nonrelational target, define the following session attributes in the **Edit Tasks** dialog box to ensure that all TIMESTAMP fields have the same format:

- Set the **Date Time Format String** to **YYYY-MM-DD HH24:MI:SS**.
- Clear **Pre 85 Timestamp Compatibility**.

Reading and Writing Binary Data in PowerExchange Client for PowerCenter

If you read binary data from any source, or if you write binary data to a DB2 for z/OS or DB2 for i5/OS target with the PowerExchange Client for PowerCenter, the data cannot be larger than 128000 bytes. Otherwise, the PowerCenter Integration Service truncates the data.

Code Pages

When you configure a connection object in the Workflow Manager, select a code page for the connection.

If you configure the PowerCenter Integration Service for code page validation, the source code page must be a code page that is a subset of the target code page.

If you configure the PowerCenter Integration Service for relaxed code page validation this is not the case. You can select any code page supported by PowerCenter for the target.

RELATED TOPICS:

- [“Oracle CDC Application Connections” on page 102](#)

APPENDIX C

PowerExchange Interfaces for PowerCenter Troubleshooting

This appendix includes the following topic:

- [Troubleshooting for PowerExchange Interfaces for PowerCenter, 219](#)

Troubleshooting for PowerExchange Interfaces for PowerCenter

When I go into Designer, I get messages about failures to load DLLs.

This can happen when PowerExchange Client for PowerCenter plug-ins are installed but cannot be loaded for various reasons like incorrect releases of PowerExchange installed or PATH problems. For more information, see [Informatica Knowledge Base Article # 15346](#).

I want to import a DB2 for i5/OS source definition, but need to determine the name of the DB2 database on the i5/OS machine.

Use the DSPRDBDIRE command to see a list of databases on the i5/OS machine.

The session failed with an error stating that the PowerExchange message repository cannot be loaded.

You can receive this error on UNIX when there is no PWX_HOME environment variable set to the PowerExchange installation directory. Set the PWX_HOME environment variable to the PowerExchange installation directory.

I set the Idle Time session condition to -1. However, the session completed with the following message: Idle Time limit is reached.

This can occur if EOF=Y is specified in the PowerExchange configuration file (dbmover.cfg) CAPI_CONNECTION statement. When you set EOF=Y, PowerExchange returns an EOF (which stops the session) when it reaches the end of the change stream as determined at the time the session starts reading from it. As a result, the PowerCenter session completes instead of continuing to run.

This message can also occur if the connection with PowerExchange is stopped through the PowerExchange STOPTASK command.

My session seems to be processing the pipelines serially.

You can configure the PowerCenter Integration Service to process master and detail pipelines sequentially. As a result, it reads data from each source in change and real-time modes sequentially.

Clear the PMServer 6.X Joiner Source Order Compatibility option on the Compatibility and Database tab in the Informatica Server Setup. When you rerun the session, the PowerCenter Integration Service will process pipelines concurrently.

The session failed with a plug-in error:

```
MAPPING> SDKS_38007 Error occurred during [initializing] reader plug-in #30nnnn.
```

This is a generic message indicating the PWXPC encountered an error. Review the session log for other messages indicating what the problem is. If there are no other error messages in the session log, check the PowerExchange logs on both the PowerCenter Integration Service platform and the Listener platform.

I want to read all of the changes I have captured and have them be inserts into a staging area. How do I do this?

When you use PowerExchange ODBC to read captured changes, INSERT is the default operation. If you want to apply the changes to the target by using the same operation as done on the source (INSERT, UPDATE, or DELETE), you need to explicitly include an Update Strategy transformation in the mapping to make this happen by testing the DTL__CAPXACTION field. In the Update Strategy Expression field, you would code:

```
DECODE (DTL__CAPXACTION, 'I', DD_INSERT, 'U', DD_UPDATE, 'D', DD_DELETE, DD_REJECT)
```

When you use PWXPC, the DTL__CAPXACTION field is automatically acted upon when processing change data. If you want to have all changes processed as INSERTs regardless of the DTL__CAPXACTION field, you must code an update strategy specifying DD_INSERT in the Update Strategy Expression field.

INDEX

A

- access method
 - CAPX [16](#)
 - CAPXRT [16, 17](#)
- Adabas
 - Batch application connection attributes for sources and targets [77](#)
 - CDC application connection attributes for sources [82](#)
 - Lookup relational connection attributes for sources [88](#)
- application name
 - restart points [117, 175](#)

B

- batch extraction mode
 - PowerExchange Condense [16](#)
- batch mode
 - configuring sessions [129](#)
- Before image
 - Flexible transactions [184](#)

C

- cache for multiple-record writes [149](#)
- CAPX
 - access method [16](#)
- CAPXRT
 - access method [16, 17](#)
- CDC application connections
 - NRDB connection attributes for nonrelational sources [82](#)
 - DB2 connection attributes for sources [70](#)
 - Microsoft SQL Server connection attributes for sources [94](#)
 - Oracle connection attributes for sources [102](#)
- CDC commit processing
 - controlling with connection attributes [122](#)
- CDC data
 - configuring lookups for [55](#)
 - group source [16, 17](#)
- CDC data map
 - extraction map [170](#)
- CDC session considerations
 - restart [172](#)
- CDC sessions
 - adding source [178](#)
 - commit processing [157](#)
 - recovery example [180](#)
 - removing source [178](#)
 - restart [117, 160](#)
 - restart token file [169](#)
 - stopping [116, 178](#)
- Change Indicator
 - Flexible transactions [184](#)
- change mode
 - configuring connections [117](#)

- change mode (*continued*)
 - configuring sessions [137](#)
- code pages
 - configuring [218](#)
 - supported code pages [218](#)
- commit processing
 - examples [124](#)
 - in CDC sessions [157](#)
 - target latency [124](#)
- compression
 - configuring [109](#)
- Condense
 - UOW Cleanser [16, 17](#)
- configuring
 - code pages [218](#)
 - compression [109](#)
 - encryption [109](#)
 - lookups for CDC data [55](#)
 - lookups for IMS [55](#)
 - pacing size [110](#)
 - sessions [129, 198](#)
 - workflows [129, 198](#)
- connection attributes
 - attributes for controlling CDC commit processing [122](#)
 - configuring Maximum Rows Per commit attribute [120](#)
 - configuring PWX Latency attribute [119](#)
 - configuring Real-Time Flush Latency attribute [119](#)
 - Event Table attribute [114](#)
 - Image Type attribute [113](#)
 - Minimum Rows Per commit attribute [121](#)
 - summary list of key attributes [108](#)
- connection attributes, PWXPC
 - UOW Count attribute [118](#)
- connections
 - target connection types [60](#)
 - configuring PWXPC connections [61](#)
 - list by source type [59](#)
- constraint-based loading
 - description [153](#)
 - FullCBLOSupport [153](#)
- continuous extraction mode
 - PowerExchange Condense [16](#)
- creating
 - DB2 source definitions [26](#)
 - DB2 target definitions [26](#)
 - IMS source definitions [32](#)
 - Oracle source definitions [30](#)
 - VSAM source definitions [32](#)
- custom property
 - FullCBLOSupport [153](#)

D

- data maps
 - non-relational source definitions [32](#)

data maps (*continued*)

viewing in the source definition [37](#)

Datacom

Batch application connection attributes for sources and targets [77](#)

CDC application connection attributes for sources [82](#)

Lookup relational connection attributes for sources [88](#)

datatypes

overview [214](#)

PowerExchange nonrelational [214](#)

PowerExchange relational [216](#)

transformation [214](#)

transformation datatypes in source qualifiers [52](#)

DB2

CDC application connection attributes for sources [70](#)

connection types [59](#)

creating source definitions [26](#)

creating target definitions [26](#)

datatypes [216](#)

DB2 Definitions [25](#)

DB2 for i5/OS

relational connection attributes for sources and targets [62](#)

DB2 for Linux, UNIX, and Windows

relational connection attributes for sources and targets [62](#)

DB2 for z/OS

relational connection attributes for sources and targets [62](#)

TIMESTAMP datatype [217](#)

DB2 stored procedure transformations

implementing in a mapping [57](#)

DB2 Stored Procedure transformations

usage considerations [56](#)

DTL__CAPXACTION

in CDC sessions [32](#), [39](#)

in extraction maps [44](#)

DTL__CAPXRESTART1

restart value [170](#), [171](#)

DTL__CAPXRESTART2

restart value [170](#), [171](#)

DTLUAPPL

example [176](#)

restart [170](#), [171](#)

DTLUTSK utility

description [178](#)

E

editing

extraction map definition [44](#)

nonrelational source definitions [39](#)

relational source definitions [32](#)

source qualifier transformations [53](#)

Source Qualifier transformations [53](#)

target definitions [32](#), [39](#)

encryption

configuring [109](#)

enhanced restart

recovery processing [180](#)

Event Table attribute

configuring [114](#)

extraction map

CDC data map [137](#)

extraction map definitions [40](#)

extraction map source definitions

editing [44](#)

viewing [42](#)

F

filelist

description [130](#)

Flexible Target Key

Custom transformations [182](#)

G

group source

CDC data [16](#), [17](#)

description [47](#)

multiple records [16](#)

sequential [16](#)

VSAM [16](#)

group target processing [50](#)

H

Hierarchy Structure metadata extension [52](#)

I

idle time

configuring for a PWXPC session [116](#)

description [116](#)

IDMS

Batch application connection attributes for sources and targets [77](#)

CDC application connection attributes for sources [82](#)

Lookup relational connection attributes for sources [88](#)

Image Type attribute

configuring [113](#)

importing

nonrelational source definitions [33](#)

nonrelational target definitions [35](#)

IMS

CDC application connection attributes for sources [82](#)

Lookup relational connection attributes for sources [88](#)

Batch application connection attributes for sources and targets [77](#)

configuring lookups for [55](#)

connection types [59](#)

datatypes [214](#)

overriding the access method and related session properties [130](#)

installing PWXPC [21](#)

L

loading

constraints [153](#)

logger token

restart value [170](#), [171](#)

lookup transformations [53](#)

Lookup transformations

usage considerations [53](#)

low values

preserving low values from source character fields [158](#)

M

mappings overview [24](#)

Maximum Rows Per commit attribute

configuring [120](#)

- metadata extensions
 - editing [39](#)
 - viewing [38](#)
 - viewing for non-relational source definitions [43](#)
- Microsoft SQL Server
 - CDC application connection attributes for sources [94](#)
 - relational connection attributes for sources and targets [91](#)
- Microsoft SQL Server definitions [28](#)
- Minimum Rows Per commit attribute
 - configuring [121](#)
- multiple-record data maps [47](#), [50](#)
- multiple-record writes
 - connection attributes [126](#)
 - considerations for sessions [149](#)
 - row statistics [151](#)

N

- non-relational source definitions
 - editing [39](#)
 - editing metadata extensions [39](#)
 - viewing data map details [37](#)
 - viewing metadata extensions [43](#)
- non-relational sources
 - configuring batch mode sessions [131](#)
- non-relational target definitions
 - editing [39](#)
 - editing metadata extensions [39](#)
- nonrelational source and target definitions [32](#)
- nonrelational sources and targets
 - CDC application connection attributes for sources [82](#)
 - Lookup relational connection attributes for sources [88](#)
 - Batch application connection attributes for sources and targets [77](#)

O

- Oracle
 - CDC application connection attributes for sources [102](#)
 - connection types [59](#)
 - creating source definitions [30](#)
 - relational connection attributes for sources and targets [99](#)
- Oracle definitions [30](#)

P

- pacing size
 - configuring [110](#)
- pipeline partitioning
 - batch mode [198](#)
 - description [156](#), [198](#)
 - loading to targets [199](#)
- PowerCenter
 - interoperability with PowerExchange [23](#)
- PowerExchange
 - performance [110](#)
- PowerExchange Change Data Capture
 - Flexible transformations [184](#)
- PowerExchange Condense
 - batch extraction mode [16](#)
 - continuous extraction mode [16](#)
- PowerExchange Configuration File
 - dbmover.cfg [22](#)
- previewing
 - PowerExchange change data [46](#)
 - PowerExchange data in Designer [44](#)

- previewing (*continued*)
 - PowerExchange nonrelational data [45](#)
 - PowerExchange relational data [44](#)
- PWX Latency attribute
 - configuring [119](#)
- PWXPC
 - installing [21](#)

R

- reader time limit
 - description for PWXPC [70](#)
- Reader Time Limit (property)
 - configuring for a PWXPC session [70](#)
- Real-Time Flush Latency attribute
 - configuring [119](#)
- real-time mode
 - configuring sessions [137](#)
- recovery
 - creating the tables [168](#)
 - enhanced restart [180](#)
 - example [180](#)
 - PM_REC_STATE table [160](#), [161](#)
 - PM_RECOVERY table [160](#)
 - PM_TGT_RUN_ID table [160](#)
 - state file [162](#)
 - tables [160](#)
- recovery strategy
 - Resume from last checkpoint [172](#)
- relational connections
 - NRDB Lookup connection attributes for nonrelational sources [88](#)
 - DB2 sources and targets [62](#)
 - Microsoft SQL Server sources and targets [91](#)
 - Oracle sources and targets [99](#)
- relational source and target definitions [25](#)
- relational source definitions
 - editing [32](#)
- relational sources
 - configuring batch mode sessions [135](#)
- relational target definitions
 - editing [32](#)
- relational targets
 - configuring sessions [143](#), [144](#)
- restart
 - \$PMRootDir/Cache [70](#), [94](#), [102](#)
 - \$PMRootDir/Restart [70](#), [94](#), [102](#), [117](#), [162](#)
 - application name [117](#), [175](#)
 - CDC sessions [117](#)
 - DTL_CAPXRESTART1 [170](#), [171](#)
 - DTL_CAPXRESTART2 [170](#), [171](#)
 - DTLUAPPL [170](#), [171](#), [175](#)
 - DTLUAPPL example [176](#)
 - earliest points [165](#)
 - logger token [170](#), [171](#)
 - null restart tokens [165](#)
 - overview [160](#)
 - PM_REC_STATE table [161](#)
 - restart token file [117](#), [162](#), [169](#)
 - restart token file folder [117](#)
 - RESTART1 [171](#)
 - RESTART2 [171](#)
 - sequence token [170](#), [171](#)
 - state file [162](#)
 - tokens [161](#), [162](#)
- restart and recover [172](#)
- restart and recovery
 - CDC session considerations [172](#)

- restart points
 - defaults if null tokens [165](#)
 - earliest [165](#)
- restart token file
 - archiving [181](#)
 - configuring [169](#)
 - example [172](#)
 - explicit override [170](#)
 - special override [171](#)
 - syntax [169](#)
- row ID
 - transformations affecting [153](#)

S

- sequence fields [50](#), [51](#)
- sequence token
 - restart value [170](#), [171](#)
- sequencing an dqueuing cache for multiple-record writes [149](#)
- session properties
 - Library/File Override property [142](#)
 - overriding the access method and related properties for IMS bulk sessions [130](#)
- sessions
 - considerations for multiple-record writes [149](#)
 - overview [129](#), [198](#)
- source definitions
 - DB2 [26](#)
 - editing metadata extensions [39](#)
 - editing, extraction maps [44](#)
 - editing, non-relational [39](#)
 - editing, relational [32](#)
 - IMS [32](#)
 - viewing metadata extensions [38](#)
 - viewing, extraction maps [42](#)
 - VSAM [32](#)
 - working with non-relational source definitions [32](#)
- source qualifier transformations
 - editing [53](#)
- source qualifiers
 - transformation datatypes [52](#)
- STOPTASK command
 - CDC sessions, stopping [116](#), [178](#)
- stored procedures
 - transformations [56](#)

T

- target definitions
 - DB2 [26](#)
 - editing metadata extensions [39](#)
 - editing non-relational [39](#)
 - editing relational [32](#)
 - viewing metadata extensions [38](#)
- terminating conditions
 - PWPXC idle time [116](#)
 - PWXPC reader time limit [70](#)
- transformations
 - affecting row ID [153](#)
 - datatypes [52](#)
 - DB2 stored procedure [56](#)
 - lookup [53](#)
 - update strategy [32](#), [39](#), [44](#)

U

- UOW Cleanser
 - Condense [16](#), [17](#)
- UOW Count attribute
 - configuring UOW Count attribute [118](#)

V

- viewing
 - extraction map definition details [42](#)
 - nonrelational source definition details [37](#)
 - nonrelational target definition details [37](#)
- VSAM
 - Batch application connection attributes for sources and targets [77](#)
 - CDC application connection attributes for sources [82](#)
 - connection types [59](#)
 - datatypes [214](#)
 - extracting data from multiple files [130](#)
 - Lookup relational connection attributes for sources [88](#)

W

- workflows
 - overview [129](#), [198](#)