Informatica® PowerExchange CDC Publisher
1.0

User Guide
# Table of Contents

Preface ................................................................. 6
Informatica Resources .................................................. 6
  Informatica Network ............................................... 6
  Informatica Knowledge Base ........................................ 6
  Informatica Documentation ......................................... 7
  Informatica Product Availability Matrixes ......................... 7
  Informatica Velocity ............................................... 7
  Informatica Marketplace ........................................... 7
  Informatica Global Customer Support ............................. 7

Chapter 1: PowerExchange CDC Publisher Overview ................. 8
Product Overview ..................................................... 8
PowerExchange CDC Publisher Data Sources and Targets .......... 9
PowerExchange CDC Publisher Architecture ........................ 10
  CDC Publisher .................................................... 10
Alternative Configurations of PowerExchange and PowerExchange CDC Publisher ................................. 11
PowerExchange CDC Publisher Utilities .............................. 16
General Data-Streaming Limitations ................................... 16

Chapter 2: Installing and Upgrading PowerExchange CDC Publisher .... 18
Installation Overview .................................................. 18
Software Requirements ................................................ 19
Installing PowerExchange CDC Publisher ........................... 20
Installed Directories .................................................... 20
Post-Installation Tasks ................................................ 21
  Configuring Environment Variables ................................ 22
  Controlling Access to PowerExchange CDC Publisher Files ... 22
Upgrading PowerExchange CDC Publisher ........................... 23
Uninstalling PowerExchange CDC Publisher .......................... 24

Chapter 3: PowerExchange CDC Publisher Key Concepts ............. 25
Publisher-Generated Messages for Data Delivery ..................... 25
Checkpointing and Guaranteed Delivery ................................ 27
Restart and Recovery .................................................. 28

Chapter 4: PowerExchange Change Capture Environment ............. 30
About the PowerExchange Change Capture Environment .......... 30
Preparing the PowerExchange Change Capture Environment ........ 31
Configuring Connectivity to PowerExchange .......................... 32
Appendix B: Avro Schema Formats

Types of Avro Schema Formats .......................... 69
Avro Fields That Define a Source Column .................. 70
Avro Flat Schema Format ...................................... 70
Avro Nested Schema Format ..................................... 71
Avro Generic Schema Format .................................... 71
Avro Wrapper Schema Format ................................... 71

Appendix C: Message Reference

Message Reference Overview ................................... 73
CDCPUB_10001 to CDCPUB_16001 ............................ 74
    CDCPUB_10001 to CDCPUB_16001 ......................... 74
CPPSvc_00100 to CPPSvc_00103 ............................... 103
LOGCFG_10001 to LOGCFG_10006 ............................. 104
PWXLIC_00110 to PWXLIC-00999 .............................. 105
PWXRDR-01000 to PWXRDR-01301 .............................. 107

Index ................................................................ 111
Preface

The *PowerExchange CDC Publisher User Guide* describes how to configure, use, and administer the PowerExchange CDC Publisher software for streaming change data to target messaging systems such as Apache Kafka. After the change data is published to the target messaging system, it can be read by consumer applications for a variety of purposes.

PowerExchange CDC Publisher is an add-on option of the Informatica PowerExchange product. The User Guide covers the following topics:

- General product architecture
- Product limitations
- Installation and upgrades
- PowerExchange CDC considerations
- Target considerations
- Configuration parameters
- Common administrative tasks
- PowerExchange CDC Publisher command-line utilities
- Supported Avro schema formats
- Messages

This guide assumes that users are familiar with the target messaging system, the PowerExchange CDC product, and the PowerExchange CDC data sources from which change data is captured.

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](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-matrices.

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.
Chapter 1

PowerExchange CDC Publisher Overview

This chapter includes the following topics:

- Product Overview, 8
- PowerExchange CDC Publisher Data Sources and Targets, 9
- PowerExchange CDC Publisher Architecture, 10
- Alternative Configurations of PowerExchange and PowerExchange CDC Publisher, 11
- PowerExchange CDC Publisher Utilities, 16
- General Data-Streaming Limitations, 16

Product Overview

The PowerExchange CDC Publisher is a Java-based tool that streams change data that has been captured from a PowerExchange data source to a target messaging system such as Apache Kafka. This tool is licensed as an option of the PowerExchange CDC product.

Typically, the PowerExchange CDC Publisher runs continuously as a Linux daemon or as a Windows foreground process until you stop it. It acts as a client of both the PowerExchange system and target messaging system. You can run the PowerExchange CDC Publisher on any Linux or Windows system in your environment, including a system that is remote from the data source, target messaging system, and PowerExchange Logger for Linux, UNIX, and Windows.

The PowerExchange CDC Publisher retrieves change data from the PowerExchange Logger log files. The CDC Publisher process creates a child extraction process that connects to the system that contains the PowerExchange Logger log files to read change data.

The PowerExchange Logger logs units of work (UOWs) in commit order. The PowerExchange CDC Publisher maintains this order when streaming change data to the target messaging system.

You can run the PowerExchange Logger in continuous mode or batch mode. However, if you configure the PowerExchange Logger to run in batch mode and stop at the "end of log," the CDC Publisher streams the change data in bursts as the batch Logger process makes the changes available in the log files.

When establishing a change data stream, the PowerExchange CDC Publisher performs the following processing:

1. Retrieves a list of extraction map names that match the schema name that you specified.
2. If you defined filtering criteria for source tables or objects, selects the extraction maps that match your filters for use in extraction processing.

3. Begins extracting change data.

4. When the first change for an extraction map is received, generates an Avro schema for the source object.

5. Formats the extracted source change records into messages based on the Avro schemas.

6. Connects to the target messaging system as a producer to send the formatted messages to target topics.

After the change data is available in the target messaging system, consumer applications, such as Informatica Intelligent Streaming, can consume the data for a variety of purposes. The consumer applications must have copies of the Avro schemas that PowerExchange CDC Publisher generated to decode the messages.

The PowerExchange CDC Publisher includes optional utilities for monitoring and administering a CDC Publisher process and for generating legible copies of the Avro schemas for consumer application use. You can run these utilities locally or from a remote system. To control access to the utility script files, you must use file system security. PowerExchange CDC Publisher does not provide security on the utility script files or administrative functions.

**PowerExchange CDC Publisher Data Sources and Targets**

PowerExchange CDC Publisher can process change data captured from relational and nonrelational PowerExchange data sources on heterogeneous platforms and stream the data to target messaging systems.

Supported source types:

- On z/OS:
  - Adabas
  - Datacom
  - DB2 for z/OS
  - IDMS
  - IMS
  - Batch VSAM
  - CICS/VSAM
  
  **Note:** PowerExchange CDC Publisher does not process data from PowerExchange Adabas, Datacom, or IDMS sources on z/OS.

- On i5/OS:
  - DB2 for i

- On Linux, UNIX, or Windows:
  - DB2 for Linux, UNIX, and Windows
  - Microsoft SQL Server (Windows only)
  - Oracle
Supported target types:

- Apache Kafka on Red Hat Linux or Windows

**PowerExchange CDC Publisher Architecture**

The PowerExchange CDC Publisher contains a Java-based CDC Publisher component that interacts with local PowerExchange client code to retrieve change data from PowerExchange. The CDC Publisher formats the change data into messages and streams the messages to the target messaging system.

The PowerExchange CDC Publisher also includes utilities for administering a CDC Publisher process and data stream, displaying the status of each process, reporting processing statistics, and generating copies of the Avro schemas for review or for consumers of the data sent to the messaging system.

The following image shows the general architecture of the PowerExchange CDC Publisher:

![PowerExchange CDC Publisher Architecture Diagram](image)

**CDC Publisher**

The Informatica CDC Publisher is a Java-based tool that is used to stream change data to a target messaging system such as Apache Kafka.

The CDC Publisher contains the following components that move data:

- The CDC Publisher Extractor consumes a stream of change data from the source. The incoming data records include schema information, row-based data changes, and transactional boundary metadata. The Extractor performs the following functions:
  - Assigns a sequence ID that is both repeatable and increasing to each change data record.
- Interacts with the component that supplies the streamed data.
- Ignores records that are older than the current restart point.
- Verifies that data is in an expected format.
- Places the results on an outbound queue for Formatter processing.

- The Filter component optionally filters the extracted change data based on lists of source objects to include or exclude that you specify.
- The Formatter receives change data from the CDC Publisher Extractor, formats the data based on the generated Avro schema of the selected format (flat, nested, or generic) for inclusion in messages, and sends the formatted messages to the Connector.
- The Connector reads the formatted messages from the Formatter and connects to the target messaging system to apply the messages. The Connector applies the message data in a consistent, ordered, and recoverable manner.

The following image shows the basic architecture of the Java-based CDC Publisher:

---

Alternative Configurations of PowerExchange and PowerExchange CDC Publisher

When determining how to implement PowerExchange CDC Publisher in your PowerExchange environment, consider the following guidelines:

- The PowerExchange CDC Publisher is a client of both the PowerExchange source system and the target messaging system. It can run on any Linux or Windows machine in your environment, including a system that is remote from the data source, target, and PowerExchange Logger for Linux, UNIX, and Windows systems.
- The PowerExchange CDC Publisher does not run on platforms other than Linux or Windows.
- PowerExchange must be installed on the source system, PowerExchange Logger system, and PowerExchange CDC Publisher system. The CDC Publisher uses the PowerExchange client code from the local PowerExchange installation to communicate with PowerExchange Listener instances on other machines.
- The PowerExchange CDC Publisher requires the PowerExchange Logger. The PowerExchange CDC Publisher reads change records from PowerExchange Logger log files. Run a PowerExchange Listener instance on the same machine as the PowerExchange Logger. Then configure the PowerExchange client.
on the CDC Publisher system to connect to the PowerExchange Listener on that machine to extract change records.

• If the PowerExchange Logger is remote from the data source, you must configure remote logging of change data, as described in the PowerExchange CDC Guide for Linux, UNIX, and Windows.

The following configurations of PowerExchange and PowerExchange CDC Publisher components are typical:

Configuration 1: All components run on the source system

The data source, PowerExchange Logger for Linux, UNIX, and Windows, PowerExchange Listener, and PowerExchange CDC Publisher run on the same Linux or Windows machine. All of the configuration files reside on this machine, including the PowerExchange dbmover configuration file, PowerExchange Logger pwxccl.cfg configuration file, all CDC Publisher configurations files, and the extraction maps. PowerExchange and the CDC Publisher can share the same dbmover configuration file.

This configuration can be used if the data source is on a Linux or Windows system. This configuration cannot be used if the data source is on an i5/OS, UNIX, or z/OS system.

Configuration 2: The PowerExchange Logger runs on the source system and the CDC Publisher runs on a separate system

You can use this configuration if the data source and PowerExchange Logger run on a Linux, UNIX, or Windows system. This configuration is also suitable when you want to add CDC Publisher functionality to an existing PowerExchange environment in which the PowerExchange Logger for Linux, UNIX, and Windows logs change data locally on the source system.

Because the PowerExchange Logger extracts change data from the data source locally, you do not need to configure remote logging. The PowerExchange client on the CDC Publisher system connects to the PowerExchange Listener on the PowerExchange Logger system to retrieve the change data. The source system and CDC Publisher systems must have separate dbmover configuration files. The dbmover configuration file on the CDC Publisher system includes a NODE statement that points to the source host.

This configuration can be used if the data source is on a Linux, UNIX, or Windows system. This configuration cannot be used if the data source is on an i5/OS or z/OS system.
The following image shows this configuration:

![Image of PowerExchange configuration diagram]

**Note:** In this image, *change data source* can be DB2 for Linux, UNIX, and Windows transaction logs, Oracle active and archived redo logs, or a Microsoft SQL Server distribution database.

**Configuration 3: The PowerExchange Logger and CDC Publisher run on a system that is remote from the source**

The PowerExchange Logger for Linux, UNIX, and Windows and PowerExchange CDC Publisher components run on the same Linux or Windows system. Because the PowerExchange Logger is remote from the data source, you must configure PowerExchange for remote logging, as described in the *PowerExchange CDC Guide for Linux, UNIX, and Windows*. The source system and the Logger and CDC Publisher system must have separate dbmover configuration files. The dbmover configuration file on the Logger and CDC Publisher system includes a NODE statement that points to the source host.

This configuration accommodates data sources on i5/OS, UNIX, or z/OS as well as data sources on Linux and Windows.
The following image shows this configuration:

![Diagram of PowerExchange CDC Publisher configuration]

**Note:** In this image, *change data source* is broadly defined as including PowerExchange Logger for z/OS log files or PowerExchange Condense files on z/OS, journal receivers or PowerExchange Condense files on i5/OS, DB2 for Linux, UNIX, and Windows transaction logs, Oracle redo logs, or a Microsoft SQL Server database.

**Configuration 4: The data source, PowerExchange Logger, and PowerExchange CDC Publisher run on three separate machines**

You can use this configuration to process change data from a source on a platform such as i5/OS or z/OS on which the PowerExchange Logger for Linux, UNIX, and Windows and PowerExchange CDC Publisher cannot run. This configuration is also suitable when you want to add CDC Publisher functionality to an existing PowerExchange environment that uses remote logging of change data to PowerExchange Logger for Linux, UNIX, and Windows log files.

On each system, define a PowerExchange dbmover configuration file. The dbmover configuration file on the CDC Publisher system includes a NODE statement that is used to connect to the source system to read extraction maps and another NODE statement that is used to connect to the Logger system to retrieve change data from the Logger log files. The CDC Publisher system also contains the PowerExchange CDC Publisher configuration files and the Kafka producer.properties file that the CDC Publisher uses to connect to Kafka.
The following image shows this configuration, including the locations of the extraction maps and the PowerExchange, CDC Publisher, and Kafka configuration files:

**Note:** In this image, *change data source* is broadly defined as including PowerExchange Logger for z/OS log files or PowerExchange Condense files on z/OS, journal receivers or PowerExchange Condense files on i5/OS, DB2 for Linux, UNIX, and Windows transaction logs, Oracle redo logs, or a Microsoft SQL Server database.

The general process flow is:

1. PowerExchange captures change data from the local data source based on capture registrations.
2. PowerExchange remotely logs the change data to the PowerExchange Logger log files on another system. The PowerExchange Logger logs changes from successful units of work (UOWs) in commit order.
3. The PowerExchange client on the PowerExchange CDC Publisher system retrieves change data from the PowerExchange Logger log files, maintaining the commit order.
4. The CDC Publisher extracts the change data for source objects of interest, formats the data into messages, and connects to the target messaging system to transmit the messages.
PowerExchange CDC Publisher Utilities

To interact with PowerExchange CDC Publisher, you must use its command-line utilities. The primary utility that initiates and processes the change data stream is the PwxCDCPublisher utility. The other utilities, PwxCDCInfo and PwxCDCAdmin, are for monitoring and administering the CDC Publisher system.

The following table briefly describes each utility:

<table>
<thead>
<tr>
<th>Utility</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>PwxCDCPublisher</td>
<td>The main PowerExchange CDC Publisher utility for streaming change data from PowerExchange to the target messaging system. The utility extracts change data from PowerExchange, filters the data if filtering criteria is defined, formats the data into messages, connects to the target, and publishes the messages to the target topics. These processes comprise a streaming path. The utility usually runs as a long-running process or daemon.</td>
</tr>
</tbody>
</table>
| PwxCDCAdmin    | The PowerExchange CDC Publisher utility for issuing administrative commands to a PwxCDCPublisher process that is running. Commands are available for:  
· Shutting down the CDC Publisher.  
· Clearing Avro schemas from cache so that you can refresh the extraction maps for source tables with added or removed columns, thereby enabling PowerExchange CDC Publisher to regenerate the Avro schemas the next time data is received for the tables.  
· Re-reading all extraction maps from PowerExchange to get new or changed extraction map definitions for source tables.  
· Reporting the current Avro schemas in a format that is suitable for viewing or consumer application use.                                                                                                                                                                                                                                                                                                                                                                                        |
| PwxCDCInfo     | The PowerExchange CDC Publisher utility for reporting the color-coded status of a PwxCDCPublisher process and each of its main subprocesses: “GREEN” for running, “RED” for stopped, or “YELLOW” for a potential problem. The utility can also report statistics and state information for each topic and for internal attributes that are intended for diagnostic use.                                                                                                                                                                                                                                                                                                                                 |
• By default, the CDC Publisher expects that the PowerExchange Logger for Linux, UNIX, and Windows is configured to extract both before-images and after-images of data. If the PowerExchange Logger is configured to extract only after images, the before images do not appear in the Avro messages. In this case, set the following properties when you configure the CDC Publisher configuration files:
  • Extract.pwxUpdateImageOption=AI
  • Formatter.avroIncludeBeforeImage=false. The setting of false disables the generation of the Avro fields for before images.

• For data sources that were registered in PowerExchange based on a COBOL copybook, the CDC Publisher uses the low-level fields but not the group that contains the fields. To handle this situation, you can use a PowerExchange expression or let the consumer application determine the method to use.

• For Oracle sources, if you specify ORACLE_UNHANDLED_NUMASCHAR=Y in the DBMOVER configuration file that the PowerExchange Navigator uses, PowerExchange converts NUMBER columns that have a precision greater than 28 or an undefined length and FLOAT columns that have a precision greater than 15 significant digits to VARCHAR(172) strings when creating capture registrations and extraction maps. The CDC Publisher does not convert the VARCHAR(172) columns back to numeric values.

• Avro represents all character strings in UTF-8 character encoding.

• If DDL changes are made to source tables, the CDC Publisher ignores the DDL changes and retains the existing message structure until you take action to update the Avro schema. For more information, see "Handling Changes to Source Tables and Extraction Maps" on page 56.
This chapter includes the following topics:

- **Installation Overview, 18**
- **Software Requirements, 19**
- **Installing PowerExchange CDC Publisher, 20**
- **Installed Directories, 20**
- **Post-Installation Tasks, 21**
- **Upgrading PowerExchange CDC Publisher, 23**
- **Uninstalling PowerExchange CDC Publisher, 24**

## Installation Overview

Informatica provides a platform-specific .zip file or .tar.gz file for installing PowerExchange CDC Publisher on a supported Linux or Windows system. To install PowerExchange CDC Publisher, you simply unzip or extract the compressed installation file.

Before you install PowerExchange CDC Publisher, perform the following tasks:

- Get the PowerExchange CDC Publisher installation file for your operating system and a PowerExchange license key that enables the PowerExchange CDC Publisher option.
- Confirm that your system meets the software requirements of PowerExchange CDC Publisher.
- Check that you have sufficient disk space on your Linux or Windows system. Informatica recommends the following minimum amounts:
  - 50 MB free disk space for the initial installation
  - 200 MB for the contents of each PowerExchange CDC Publisher instanceX subdirectory that you configure

After you install PowerExchange CDC Publisher, you must perform some initial system-setup tasks to prepare the PowerExchange CDC Publisher environment, such as configuring environment variables.
Software Requirements

Verify that the system where you plan to install PowerExchange CDC Publisher meets the product operating system and software requirements.

Tip: The Product Availability Matrix (PAM) for this product version provides the latest information on supported versions of operating systems and software. If you are an Informatica Network member, you can access the current PAM at https://network.informatica.com/community/informatica-network/product-availability-matrices.

Supported Operating Systems

PowerExchange CDC Publisher is certified to run on the following operating systems:

- Red Hat Enterprise Linux versions 6.5 and 7.0
- Windows 10
- Windows Server 2012 R2
- Windows Server 2008 R2

Supported PowerExchange Versions

PowerExchange CDC Publisher works with the following PowerExchange CDC versions:

- 10.2
- 10.1.1
- 10.1

For supported versions of the PowerExchange data sources, see the PowerExchange Installation and Upgrade Guide for your PowerExchange version.

Supported Apache Kafka Target Versions

PowerExchange CDC Publisher works with the following Apache Kafka versions:

- 0.10.2 from any available binary download
- 0.9.0.1 from any available binary download

Note: The Scala version is immaterial.

Other Software Requirements

On the PowerExchange CDC Publisher system, ensure the following software is also installed:

- 64-bit Java JRE 8 (root directory jre1.8.0_version) or later
- Microsoft Visual C++ Redistributable Packages for Visual Studio 2013 (Windows only)

Note: The PwxCDCPublisher script looks for the msvcr120.dll file in the Windows path in the Path environment variable.
Installing PowerExchange CDC Publisher

To install PowerExchange CDC Publisher on a 64-bit Linux or Windows system, you simply extract the compressed installation file that Informatica delivers for the operating system in the pwxcdcpublisher_100_cd.zip file.

1. Transfer the platform-specific installation file that you received from Informatica to a directory on your Windows or Linux system.
   - Informatica provides the pwxcdcpubv.r.m_win_x64.zip file for the Windows installation and the pwxcdcpubv.r.m_linux_x64.tar.gz file for the Linux installation. Retain the installation file so that you can use it again if you need to reinstall the PowerExchange CDC Publisher.

2. Unzip or extract the installation file to a location of your choice.
   - The PowerExchange CDC Publisher root installation directory and subdirectories are created. The root installation directory has a unique version-specific name that will not be overwritten by subsequent upgrade installations.

3. Verify that the expected PowerExchange CDC Publisher directories and files are present.

Installed Directories

After you install PowerExchange CDC Publisher, the following directories and subdirectories exist:

```
pwxcdcpub_root_installation_directory
    bin
    instanceA
        checkpoint
        config
        logs
        overflow
        reports
        libs
        samples
```

The following table describes these directories:

<table>
<thead>
<tr>
<th>Directory</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>pwxcdcpub_root_installation_directory</td>
<td>The product root installation directory that is created when you unzip or extract the compressed installation file for your operating system. The installation provides a version-specific root installation directory name. You can optionally change it to another unique name.</td>
</tr>
<tr>
<td>bin</td>
<td>Contains the script files for running the PowerExchange CDC Publisher utilities: PwxCDCAdmin, PwxCDCInfo, and PwxCDCPublisher.</td>
</tr>
<tr>
<td>instanceA</td>
<td>Contains the checkpoint, config, logs, overflow, and reports subdirectories for the default instance of PowerExchange CDC Publisher. You can create additional instance directories under other names such as instanceB or MyInstance, each with their own checkpoint, config, logs, overflow, and reports subdirectories.</td>
</tr>
<tr>
<td>Directory</td>
<td>Description</td>
</tr>
<tr>
<td>-----------</td>
<td>-------------</td>
</tr>
<tr>
<td>checkpoint</td>
<td>Contains the checkpoint files that record the sequence number of the last change operation that was successfully sent to the target messaging system. These files are used to restart apply processing from the last change operation processed, thereby avoiding missing or duplicate changes on the target.</td>
</tr>
<tr>
<td>config</td>
<td>Contains the customized PowerExchange CDC Publisher configuration files that the PowerExchange CDC Publisher utilities use. Also contains the Log4j logging configuration file, PwxCDCPubLog4j.xml file, that configures the logging of PowerExchange CDC Publisher messages. You must copy the configuration files from the samples directory to this config directory to customize and use the files.</td>
</tr>
<tr>
<td>logs</td>
<td>Contains the active and archived log files with the PowerExchange CDC Publisher message output. The log files also contain PowerExchange error messages and optionally some PowerExchange informational messages.</td>
</tr>
<tr>
<td>overflow</td>
<td>Contains PowerExchange CDC Publisher cache files in temporary storage.</td>
</tr>
<tr>
<td>reports</td>
<td>Contains output from PwxCDCAdmin Report command. This command reports the Avro schemas that have been generated for source tables. These schemas can be used by consumer applications.</td>
</tr>
<tr>
<td>libs</td>
<td>Contains the PowerExchange CDC Publisher shared libraries that include the .jar files and the .dll or .so files.</td>
</tr>
<tr>
<td>samples</td>
<td>Contains the sample PowerExchange CDC Publisher configuration files. <strong>Tip:</strong> Informatica recommends that you retain the sample files &quot;as is&quot; so that they are available for re-use. Copy the files to the config subdirectory of the instance you intend to configure and then customize the file copies there.</td>
</tr>
</tbody>
</table>

### Post-Installation Tasks

You must prepare some initial setup tasks to prepare the system for PowerExchange CDC Publisher.

Perform the following tasks before running Powerexchange CDC Publisher:

- Set the environment variables that PowerExchange CDC Publisher uses.
- Ensure that the PWX_LICENSE environment variable points to the correct license.key file.
- Configure file system security to control access to PowerExchange CDC Publisher files, as needed.
Configuring Environment Variables

After you install PowerExchange CDC Publisher, verify that the required environment variables are correctly set.

The following table summarizes the environment variable settings for a CDC Publisher environment:

<table>
<thead>
<tr>
<th>Environment Variable</th>
<th>Setting</th>
</tr>
</thead>
<tbody>
<tr>
<td>JAVA_HOME</td>
<td>The path to the base directory for your 64-bit JRE installation on the PowerExchange CDC Publisher system, for example: C:\Program Files\java\jre1.8.0_144.</td>
</tr>
<tr>
<td>KAFKA_CLIENT_LIBS</td>
<td>The path to the Apache Kafka client libs directory, which contains the Kafka .jar files, on the PowerExchange CDC Publisher system, for example: C:\kafka_2.11-0.11.0.0\libs.</td>
</tr>
<tr>
<td>PWX_CONFIG</td>
<td>The path and file name of the PowerExchange DBMOVER configuration file on the system where the PowerExchange CDC Publisher runs.</td>
</tr>
<tr>
<td>PWX_LICENSE</td>
<td>The path and file name of the PowerExchange license key file that enables use of the PowerExchange CDC Publisher option for the target messaging system. <strong>Note:</strong> You can change the path and file name of the license key file that is delivered with the PowerExchange CDC Publisher, provided that you specify the new path and file name in this environment variable.</td>
</tr>
<tr>
<td>PWXPUB_HOME</td>
<td>The path to the PowerExchange CDC Publisher root installation directory, for example: C:\pwxcdcpub\bin\bin\subdirectory. Define this environment variable if you plan to run the PowerExchange CDC Publisher utility scripts by using batch files instead of running the scripts directly from the root installation directory. In each batch file, you can reference %PWXPUB_HOME% instead of the full path to the root installation directory.</td>
</tr>
</tbody>
</table>

Controlling Access to PowerExchange CDC Publisher Files

PowerExchange CDC Publisher relies on file system security to control access to files and scripts. You are responsible for configuring this type of security on your operating system.

To protect against unauthorized use of any PowerExchange CDC Publisher script in the pwxcdcpub_root_installation\bin subdirectory, configure file-system privileges to control who has access to these scripts. For example, to provide userA with permission to run the PwxCDCInfo script but not run the PwxCDCAdmin or PwxCDDCPublisher script, you could use the following strategy:

1. Grant read-only access to all files in the PowerExchange CDC Publisher installation to userA, except the script files in the bin subdirectory.
2. Copy the PwxCDCInfo script to another directory and grant execute permissions on that instance of the script to userA.
3. Set the PWXPUB_HOME environment variable for userA to point to the PowerExchange CDC Publisher root installation directory.

This environment variable setting is required to run PowerExchange CDC Publisher scripts from a location other than the pwxcdcpub_root_installation\bin subdirectory.

For more information about setting up file-system security, contact your Security Administrator.
Upgrading PowerExchange CDC Publisher

If PowerExchange CDC Publisher is currently installed, you can use this procedure to upgrade to a new PowerExchange CDC Publisher version.

The upgrade procedure adds the new installation without deleting or overwriting the previous installation. This strategy allows you to copy the information defined for your existing instances to the new installation incrementally, as needed. Also, if you need to fall back to the older version, it is still available until you uninstall it.

Before you upgrade the product, perform the following tasks:

- Read the PowerExchange CDC Publisher Release Notes to determine if any upgrade issues exist.
- Verify that your system is compatible with the software requirements of the new version.

1. Download the PowerExchange CDC Publisher installation file for your operating system to a directory other than the current PowerExchange CDC Publisher installation directory.
   
   Tip: Retain the installation file so that you can use it to upgrade other PowerExchange CDC Publisher installations if necessary.

2. Shut down the PwxCDCPublisher process if it is running. To do so, run the PwxCDCAdmin command with the SHUTDOWN parameter.
   
   Note: You do not need to stop PowerExchange CDC processing or Apache Kafka.

3. Unzip or extract the installation file for the new PowerExchange CDC Publisher version. Typically, the file is extracted to a directory other than the existing PowerExchange CDC Publisher installation directory.
   
   The software is extracted into a version-specific root installation directory so that it does not overwrite your previous installation. You can rename the new installation directory, for example, from "pwxcdcpubvr64_win_x64" to "PowerExchangeCDCPublisher_vrm." Alternatively, you can rename the previous installation directory and then use its original name as the name of the new installation so that you do not have to change the PWXPUB_HOME environment variable value, if set.

4. If you defined the PWXPUB_PUB environment variable, ensure that the environment variable points to the new installation.

5. If you have a new license key file, ensure that the PWX_LICENSE environment variable points to the new file.

6. Copy the instanceX directories that you use from the previous installation to the new installation. Confirm that all of the subdirectories under each instance are also copied, including the config subdirectory that contains the PowerExchange CDC Publisher configuration files.
   
   Note: You can copy the instanceX directories to the new installation incrementally, as they are needed.

7. If the new CDC Publisher version contains new or changed configuration properties, update the CDC Publisher configuration files in the config subdirectory as needed.
   
   To determine if CDC Publisher properties changed, see the PowerExchange CDC Publisher documentation.

8. To verify that the upgrade succeeded, restart the PowerExchange CDC Publisher process and then run the PwxCDCInfo command with the STATUS parameter.
   
   By default, the Publisher process starts from the last checkpoint position. If you want to start the Publisher process from a different position, issue the PwxCDCPublisher command with the RESTART parameter. For more information, see "PwxCDCPublisher Utility - Command and Parameters" on page 63.
   
   For information about running the PwxCDCInfo command with the STATUS parameter, see "PwxCDCInfo Utility - Command and Parameters" on page 64.
Uninstalling PowerExchange CDC Publisher

To remove PowerExchange CDC Publisher from your system, you can simply delete the top-level installation directory.

1. Shut down the PowerExchange CDC Publisher process by running the PwxCDCAdmin utility with the SHUTDOWN parameter.
2. Delete the PowerExchange CDC Publisher root installation directory.
3. Verify that the PowerExchange CDC Publisher root installation directory and all of its subdirectories and files have been removed.
4. Optionally, remove the environment variables that you added for PowerExchange CDC Publisher.
This chapter includes the following topics:

- Publisher-Generated Messages for Data Delivery, 25
- Checkpointing and Guaranteed Delivery, 27
- Restart and Recovery, 28

Publisher-Generated Messages for Data Delivery

PowerExchange CDC Publisher creates a message for each change record that it receives for a source table or object and sends the message to the target messaging system. The format of the message is based on an Avro schema that the PowerExchange CDC Publisher generates. To generate an Avro schema, the PowerExchange CDC Publisher uses the PowerExchange extraction map for the source object.

Note: PowerExchange CDC Publisher does not generate messages for DDL operations or for UOW begin and commit records.

Message Content

A message contains extracted change data and metadata. The following table describes the fields that are included in a generated message:

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>DTL__xxx</td>
<td>The PowerExchange-generated DTL__ columns that have been added to extraction maps by default or by the PowerExchange user. For more information about these columns, see the PowerExchange Navigator User Guide.</td>
</tr>
<tr>
<td>INFA_OP_TYPE</td>
<td>The change operation type (INSERT, UPDATE, or DELETE) that was extracted from the source.</td>
</tr>
<tr>
<td>INFA_TABLE_NAME</td>
<td>The source mapname_tablename from the extraction map name. This value identifies the source object for which change data was extracted.</td>
</tr>
<tr>
<td>INFA_SEQUENCE</td>
<td>A sequence number that the CDC Publisher assigns to the change record.</td>
</tr>
</tbody>
</table>
### Field Description

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>source_column_name</code></td>
<td>The after image of a source column to which a change operation was applied.</td>
</tr>
<tr>
<td><code>source_column_name_Present</code></td>
<td>An indicator of whether the column contains a value from a change operation.</td>
</tr>
<tr>
<td><code>source_column_name_BeforeImage</code></td>
<td>The before image value of an updated source column.</td>
</tr>
<tr>
<td><code>source_column_name_BeforeImage_Present</code></td>
<td>An indicator of whether the before image of an updated source column is present.</td>
</tr>
</tbody>
</table>

For UPDATE operations, the Avro messages include both the before image and after image. You can set the `Formatter.avroIncludeBeforeImage` property to false in the `cdcPublisherAvro.cfg` configuration file to not generate Avro fields for before images.

**Note:** If you specify `CAPTURE_IMAGE=AI` in the PowerExchange Logger `pwxccl.cfg` file to capture after images only, no before image data will be available.

### Message Format and Encoding

PowerExchange CDC Publisher produces messages only in Avro format with the encoding type that you select.

You control the Avro encoding by setting the `Formatter.avroEncodingType` configuration property in the `cdcPublisherAvro.cfg` configuration file. You can specify an encoding type of **JSON**, **binary**, or **none**. The CDC Publisher uses the specified encoding type when serializing records in an Avro format. Set the encoding type to **none**, which indicates no explicit encoding, if you use a third-party encoding schema, such as the Confluent Schema Registry encoding schema.

The PowerExchange CDC Publisher generates an Avro schema when it receives change data for a source table or object. The PowerExchange CDC Publisher can generate Avro schemas in one of the following formats, depending on how you set the `Formatter.avroSchemaFormat` property in the `cdcPublisherAvro.cfg` configuration file:

- **Flat.** Lists all Avro fields in one Avro record. A unique Avro schema is generated for each source object.
- **Nested.** Organizes each type of information in a separate Avro record. A unique Avro schema is generated for each source object.
- **Generic.** Generates an Avro schema that accommodates more than one source object. The source column names are included in each record, which allows the generic schema to be independent of any source object. All of the PowerExchange-generated `DTL__xxx` columns are included as metadata.

Additionally, you can "wrap" a flat, nested, or generic schema with an Avro schema that acts as a header that contains metadata followed by the underlying Avro schema for the source object. To do so, set the `Formatter.avroWrapperSchemaFormat` property to `avroWrapperSchemaFormatV1` in the `cdcPublisherAvro.cfg` file. The wrapper schema format contains the following Avro fields:

- **Sequence number of the change record**
- **Change operation type**
- **Source mapname tablename** from the extraction map name
- The "wrapped" Avro schema of the type specified in the `Formatter.avroSchemaFormat` property

All messages based on the wrapper schema have the same four-field format.
You can use a generic or wrapper schema to represent multiple source tables. The consumer of data in the target topics examines the metadata to determine which source table is represented and discover the source table structure. The consumer can then parse the underlying data and put it into the proper Avro format. Consider using a generic or wrapper schema when you want to send messages with change data from multiple source objects to a single Kafka topic. The topic is identified in the Connector.kafkaTopic property.

A flat or nested schema pertains to a specific source table. It defines the columns in the source table. Consider using a flat or nested schema when you want to send change data from a specific source object to the Kafka topic that is generated for that source object. In this case, set the Connector.kafkaTopic property to USE_TABLE_NAME.

**Supported Avro Types**

The Avro schemas that the CDC Publisher generates support all Avro primitive types except FLOAT or DOUBLE, which are represented as strings. Also, if you use Avro logical types for dates, decimal values, times, or timestamps, the CDC Publisher makes a best-effort attempt to process the logical types under the following conditions:

- A Formatter.avroUseLogicalType property is set to true in the cdcPublisherAvro.cfg configuration file, where type is DateType, DecimalType, TimeMillisType, TimeMicrosType, TimestampMillisType, or TimestampMicrosType.
- The source field is defined in the extraction map with a compatible data type, scale, and precision.

**Note:** The CDC Publisher does not convert a timestamp to a date type.

The Publisher-generated Avro schemas do not support complex types.

**Avro Schema Usage Considerations**

Consumer applications that read data from the messaging system must have a copy of the Avro schema to decode the message. You can use the PwxCDCAdmin utility to generate Avro schemas in a legible format that consumer applications can use. For more information, see "Reporting the Avro Format Definitions for Source Tables" on page 56.

If you change the structure of a source table or object and update the extraction map, or if you change any of the Avro-related properties in the cdcPublisherAvro.cfg configuration file, the PowerExchange CDC Publisher does not automatically update any existing Avro schema. You can use the PwxCDCAdmin utility to clear the existing Avro schema from cache and then regenerate the Avro schema. For more information, see "Handling Changes to Source Tables and Extraction Maps" on page 56.

**Checkpointing and Guaranteed Delivery**

PowerExchange CDC Publisher generates a checkpoint file after it sends the first change operation. As data streaming progresses, the CDC Publisher saves information about the last change operation processed to the checkpoint file. This checkpoint information is used to resume CDC Publisher apply processing after the CDC Publisher is restarted.

The CDC Publisher uses only one checkpoint file for each instance. By default, the file is named “checkpoint” and is created in the “checkpoint” subdirectory of an instance. You can change the file name and directory by specifying the Connector.kafkaFileCheckpointFileName and Connector.kafkaCheckpointFileDirectory properties in the cdcPublisherKafka.cfg configuration file.

The checkpoint file contains checkpoint information only for the last change operation processed. The checkpoint format is specific to the CDC Publisher and should not be edited. When a checkpoint is written depends on whether you set the Connector.kafkaProducerGuaranteeDelivery property to false in the cdcPublisherKafka.cfg file or accept the default value of true. With the default value of true, the CDC
Publisher uses *guaranteed delivery* to write a checkpoint after each change operation is acknowledged as successfully received by the target messaging system. This delivery mode avoids message loss and duplication but slows apply processing. If you set this property to false and the target messaging system terminates while changes are in flight, the CDC Publisher will not skip any change operations but might apply duplicate messages to the target messaging system after the change stream is restarted.

The following considerations pertain to using checkpoints and checkpoint files:

1. If connectivity to the target messaging system is lost or the CDC Publisher terminates, the CDC Publisher process will restart from the checkpoint position that is recorded in the checkpoint file by default. In this situation, some messages might be duplicated on the target messaging system. To guarantee that messages are not duplicated, ensure that the Connector.kafkaProducerGuaranteeDelivery property is set to true.
2. If the PowerExchange CDC Publisher process ends abnormally, the checkpoint value in the checkpoint file might not be accurate in the following situations:
   - The existing checkpoint value does not reflect the last change operation because the change stream terminated after the CDC Publisher sent a message to the Kafka target and before Kafka acknowledged the message as received. In this case, the CDC Publisher still restarts from the existing checkpoint value. Some messages that were previously sent to the target messaging system might be resent.
   - The checkpoint file is corrupted. This situation can occur if an attempt to write a checkpoint value to the checkpoint file failed or did not complete. In this case, delete the checkpoint file. Then configure a restart point by setting the Extract.restart1 and Extract.restart2 properties in the cdcPublisherPowerExchange.cfg file. When you restart the CDC Publisher, use the RESTART=FROM_CONFIG parameter. If you do not configure a specific restart point, the CDC Publisher restarts from the oldest point in the log files, as if RESTART=FROM_BEGINNING is specified.

## Restart and Recovery

You might need to restart a PowerExchange CDC Publisher process under normal conditions or after a CDC Publisher process, PowerExchange component, or Apache Kafka target terminates unexpectedly.

The following list describes the default restart behavior under these circumstances:

1. If you restart the CDC Publisher process under normal conditions or after a PowerExchange Logger or Listener failure, the CDC Publisher process restarts from the last change operation sent, as recorded in the checkpoint file, by default.
2. If you restart the CDC Publisher process after it ends abnormally, it restarts from the last change operation sent, as recorded in the checkpoint file, by default. If a change operation was sent but has not been acknowledged as received by Kafka, the change record is resent.
3. If you restart the CDC Publisher process after the Kafka target messaging system ends abnormally, CDC Publisher apply processing restarts based on the last change operation recorded in the checkpoint file by default. If you disabled guaranteed delivery by specifying Connector.kafkaProducerGuaranteeDelivery=No in the cdcPublisherKafka.cfg configuration file, some changes might be resent to the target. If guaranteed delivery is enabled, no changes are duplicated or missed.

You can override the default CDC Publisher restart behavior by entering the PwxCDCPublisher command with the RESTART parameter at the command line. You can configure the RESTART parameter to restart the CDC Publisher process from the earliest available change record in the PowerExchange Logger log files (FROM_BEGINNING) or from a specific position in the change stream (FROM_CONFIG), which you specify in the Extract.restart1 and Extract.restart2 properties in the cdcPublisherPowerExchange.cfg file. For more information, see "[Restarting a PowerExchange CDC Publisher Change Data Stream](#) on page 60."
Note: To cold start the CDC Publisher process FROM_BEGINNING, you must delete the checkpoint file. Otherwise, all change operations prior to or equivalent to the checkpointed change operation are not extracted and sent to the target.
CHAPTER 4

PowerExchange Change Capture Environment

This chapter includes the following topics:

- About the PowerExchange Change Capture Environment, 30
- Preparing the PowerExchange Change Capture Environment, 31
- Configuring Connectivity to PowerExchange, 32

About the PowerExchange Change Capture Environment

Before you can run PowerExchange CDC Publisher processes, you must perform some prerequisite tasks in PowerExchange and configure connectivity to PowerExchange. This section summarizes these tasks.

For information about configuring PowerExchange to capture change data from data sources, see the Powerexchange documentation set for your PowerExchange version.
You must perform some PowerExchange configuration tasks for the PowerExchange CDC Publisher, even if you are already using PowerExchange to capture change data from data sources.

Perform some or all of the following configuration tasks:

<table>
<thead>
<tr>
<th>Task</th>
<th>Reference</th>
</tr>
</thead>
<tbody>
<tr>
<td>On the CDC Publisher machine, install a PowerExchange version that the CDC Publisher supports, unless you will run the CDC Publisher on an existing PowerExchange Navigator or PowerExchange Logger system.</td>
<td>Chapter “Installing and Upgrading PowerExchange on Linux and UNIX” or “Installing and Upgrading PowerExchange on Windows” in the PowerExchange Installation and Upgrade Guide</td>
</tr>
<tr>
<td>Ensure that a local copy of the PowerExchange license key file that includes the PowerExchange CDC Publisher option for the target messaging system exists. This PWX_LICENSE environment variable should point to this file.</td>
<td>-</td>
</tr>
<tr>
<td>In the PowerExchange Navigator, if capture registrations and extraction maps for the source tables or objects do not yet exist, create a capture registration for each source table or object. In the new registrations, set the Status option to Active and the Condense option to Part. The PowerExchange Navigator generates an extraction map for each capture registration that you create.</td>
<td>Chapters &quot;Registration Groups and Capture Registrations&quot; and &quot;Extraction Groups and Extraction Maps&quot; in the PowerExchange Navigator User Guide</td>
</tr>
<tr>
<td>On the PowerExchange CDC Publisher system, configure a dbmover configuration file that includes the following statements: - A NODE statement that points to the PowerExchange Listener where the PowerExchange Logger for Linux, UNIX, and Windows runs. This node name must match the Extract.pwxNodeLocation property in the cdcPublisherPowerExchange.cfg file. If you use remote logging of change data to PowerExchange Logger log files, the node name must also match the CAPTURE_NODE value in the PowerExchange Logger configuration file, pwxccl.cfg. - If the extraction maps and PowerExchange Logger are not on the same system, define another NODE statement that points to the system that contains the extraction maps. This node name must match the Extract.pwxXmapLocation property value in the cdcPublisherPowerExchange.cfg file. - If the PowerExchange Listener or PowerExchange Logger is not on the same system as the CDC Publisher, optionally include a set of SVCNODE and CMDNODE statements for the Listener and for the Logger to be able to issue commands to these PowerExchange components from the local CDC Publisher system.</td>
<td>Chapter “DBMOVER Configuration” in the PowerExchange Reference Manual and the source-specific chapter in the PowerExchange CDC guide for your source platform.</td>
</tr>
</tbody>
</table>
Configuring Connectivity to PowerExchange

The PowerExchange CDC Publisher must be able to connect to the system where the PowerExchange Logger for Linux, UNIX, and Windows log files reside and to the system with the source extraction maps. To configure connectivity to PowerExchange, you must specify certain PowerExchange dbmover statements and CDC Publisher properties.

DBMOVER Configuration Statements

If the CDC Publisher and PowerExchange Logger are on the same system, they can share the same local PowerExchange dbmover configuration file.

If the CDC Publisher is remote from the PowerExchange Logger or source system, you must create a dbmover configuration file on the CDC Publisher system. In this dbmover configuration file, configure one or both of the following connectivity statements:

- If the CDC Publisher system is remote from the PowerExchange Logger system, define a NODE statement that points to the PowerExchange Logger host system.
- If the CDC Publisher is remote from the source system where the extraction maps reside, define a NODE statement that points to the source host with the extraction maps.

Also ensure that the PWX_CONFIG environment variable points to the local dbmover configuration file on the CDC Publisher system.

For more information about the NODE statement, see the PowerExchange Reference Manual.

CDC Publisher Configuration Properties

In the cdcPublisherPowerExchange.cfg file on the CDC Publisher system, specify the following properties:

- If the location of the extraction maps is different from the Extract.pwxNodeLocation location from which change data will be extracted, specify the Extract.pwxXmapLocation property. In this property, specify the node name from the NODE statement that points to the location of the extraction maps. If a user ID and password are required to access this node, also specify the Extract.pwxXmapUserId and Extract.pwxXmapPwd properties.
- In the Extract.pwxNodeLocation property, specify the node name from the NODE statement that points to the PowerExchange Listener on the system from which change data will be extracted. If a user ID and
password are required to access this node, also specify the Extract.pwxNodeUserld property and the Extract.pwxNodePwd or Extract.pwxNodeEPwd property.

- In the Extract.pwxCapiConnectionName property, specify the NAME value from the CAPX CAPI_CONNECTION statement that is in the dbmover configuration file on the PowerExchange Logger system. This statement is used to continuously extract change data from PowerExchange Logger log files.

For more information about these properties, see "PowerExchange Extract Configuration Properties" on page 41.
Chapter 5

Apache Kafka Targets

This chapter includes the following topics:

- About Kafka Targets, 34
- Preparing the Apache Kafka Environment, 34
- Apache Kafka Data-Streaming Considerations, 35

About Kafka Targets

Apache Kafka is an open-source distributed data streaming platform that is scalable and fault tolerant and able to process large amounts of data quickly. It can serve as an interim staging area for data that will be consumed by various types of downstream consumer applications.

Producer applications, such as the PowerExchange CDC Publisher, can publish streams of change records as messages to Kafka topics. Kafka makes the messages available with low latency to consumer applications that subscribe to the message topics based on the topic names.

For information about Kafka use cases, configuration, and implementation, see the Apache Kafka documentation at https://kafka.apache.org/.

Preparing the Apache Kafka Environment

You must perform a few tasks on the PowerExchange CDC Publisher and Kafka target systems to prepare the Apache Kafka environment for use with PowerExchange CDC Publisher.

1. Download the Kafka client libraries to the PowerExchange CDC Publisher system.
   The version of the Kafka client libraries must match the version of the Kafka target. The Kafka client libraries do not need to be copied under the PowerExchange CDC Publisher root installation directory.

2. On the PowerExchange CDC Publisher system, define the KAFKA_CLIENT_LIBS environment variable.
   This environment variable points to the libs subdirectory of the local Kafka client installation. The libs directory contains Kafka .jar files.

3. Ensure that the CDC Publisher can access the Kafka producer.properties file. The file can be on the local CDC Publisher host or a shared device.
This file contains connectivity information that CDC Publisher Connector will use to communicate with the Kafka target.

You can optionally rename the file name. When you configure the PowerExchange CDC Publisher configuration files later, you will set the Connector.kafkaProducerPropertiesFile property to point to this producer.properties file.

4. On the Kafka target system, if Kafka is not configured to automatically generate Kafka topics, create the Kafka topic or topics to which the PowerExchange CDC Publisher will publish the messages that contain change data. Ensure that the topics are created before you run the PowerExchange CDC Publisher process.

You can create a single topic for all source tables or create a separate topic for each source table. When you configure the PowerExchange CDC Publisher configuration files later, set the Connector.kafkaTopic property to specify either the name of single topic to which all messages will be published or the option USE_TABLE_NAME to publish messages to a separate topic for each source table.

**Note:** By default, the auto.create.topics.enable parameter in the Kafka server.properties file is set to true. With this setting, the Kafka broker automatically generates the topic or topics if they do not exist the first time messages are sent to the topics. You need to manually create the topics only if the auto.create.topics.enable parameter has been set to false.

For more information about configuring a Apache Kafka system, see the Apache Kafka documentation.

### Apache Kafka Data-Streaming Considerations

The following data-streaming considerations and limitations apply to Kafka targets:

- The Kafka producer.properties file controls PowerExchange CDC Publisher connectivity to Kafka.
- The PowerExchange CDC Publisher supports Kafka primitive datatypes except FLOAT and DOUBLE. Also, the CDC Publisher tries to convert logical types for dates, times, timestamps, and decimal values on a best effort basis if you set the Formatter.avroUseLogicalType property to true in the cdcPublisherAvro.cfg configuration file, where type is Date, Decimal, TimeMillis, TimeMicros, TimestampMillis, TimestampMicros. The CDC Publisher does not support complex types.

**Note:** The FLOAT and DOUBLE types are not supported because the CDC Publisher converts all source data to characters in Kafka messages. Any subsequent attempt to convert the character data to FLOAT or DOUBLE would decrease data accuracy and precision.

- To minimize the risk of duplicate or missing messages on the Kafka target after a network outage, ensure that the Connector.kafkaProducerGuaranteeDelivery property is enabled. This property is enabled by default. Guaranteed delivery provides the highest level of data integrity but might degrade performance. You can disable guaranteed delivery by setting the Connector.kafkaProducerGuaranteeDelivery property to false in the cdcPublisherKafka.cfg configuration file.
CHAPTER 6

Configuring PowerExchange CDC Publisher

This chapter includes the following topics:

- Configuration Overview, 36
- Customizing the PowerExchange CDC Publisher Configuration Files, 37
- Filtering Extraction Maps and Source Objects, 44
- Configuring CDC Publisher Message Logging, 46
- If Informatica Intelligent Streaming Will Consume Data from Kafka, 46

Configuration Overview

After you install PowerExchange CDC Publisher and prepare the PowerExchange source environment and the target messaging system, configure the parameters in the PowerExchange CDC Publisher configuration files according to your data streaming requirements.

By default, under the PowerExchange CDC Publisher root installation directory, a single instanceA directory exists and contains checkpoint, config, logs, overflow, and reports subdirectories. You can create additional instance subdirectories, each with a unique directory name and separate set of subdirectories and configuration files. Consider using multiple instances if you want to process changes from a large number of source objects in multiple data streams. In this case, configure each instance to use a different set of extraction maps to process changes for different source objects. When you run the PowerExchange CDC Publisher process, specify which instance to use in the INSTANCE parameter.

For each instance that you create, perform the following configuration tasks:

1. Copy all of the PowerExchange CDC Publisher configuration files from the \pwxcdcpub_root_installation\samples subdirectory to the \pwxcdcpub_root_installation \instanceX\config subdirectory. PowerExchange CDC Publisher uses the configuration files that are in the config subdirectory.

2. Customize the properties in the PowerExchange CDC Publisher configuration files that are in the config subdirectory. See "Customizing the PowerExchange CDC Publisher Configuration Files" on page 37.

3. If you do not want to stream change data based on all of the extraction maps that match the extraction-map schema name that you specify, enter filtering criteria in the Extract.captureConsumerExcludeNameList property, Extract.captureConsumerIncludeNameList property, or both in the cdcPublisherPowerExchange.cfg configuration file. See "Filtering Extraction Maps and Source Objects" on page 44.
4. Customize the Log4j 2 logging of PowerExchange CDC Publisher messages, if necessary. Usually, the sample PwxCDCPubLog4j.xml configuration file that PowerExchange CDC Publisher supplies can be used without modification. See "Configuring CDC Publisher Message Logging" on page 46.

5. If the Informatica Intelligent Streaming product will consume the messages with change data that the PowerExchange CDC Publisher sends to a Kafka target, configure the Formatter parameters for treating fields with a binary datatype as string data and for using binary-encoded messages. See "If Informatica Intelligent Streaming Will Consume Data from Kafka" on page 46.

Customizing the PowerExchange CDC Publisher Configuration Files

PowerExchange CDC Publisher provides sample configuration files in the pwxcdcpub_root_installation/samples subdirectory. These files contain the basic CDC Publisher properties that you will need to set. For special circumstances, Informatica technical staff might request that you set additional properties.

The following sample configuration files are provided:

- *cdcPublisherAvro.cfg*. Contains properties that specify the Avro schema format and the data serialization method and encoding for Kafka messages.
- *cdcPublisherCommon.cfg*. Contains properties that define the PowerExchange CDC Publisher host and port.
- *cdcPublisherKafka.cfg*. Contains properties related to Apache Kafka targets.
- *cdcPublisherPowerExchange.cfg*. Contains properties related to accessing PowerExchange extraction maps and change data.

You can copy the sample configuration files to the pwxcdcpub_root_installation/instanceX/config subdirectory and then customize the copies of the files. The files contain comments that describe each property. The PowerExchange CDC Publisher will read only the configuration files that are in the config subdirectory.

For PowerExchange CDC Publisher to recognize a file as a configuration file, the file must have the file name extension *.cfg*. Also, the file content must begin with "Configuration on the first or second line in the file.

You can include all configuration properties on one file or create alternative subsets of properties in multiple configuration files. Informatica recommends that you use multiple configuration files that logically group the properties, similar to the sample configuration files. You can then copy and re-use the properties that do not change across multiple instances and migration paths, and customize only the properties that need to change because of changes to your CDC Publisher environment. For example, you might want to customize the PowerExchange-related extraction properties and Kafka target properties when migrating PowerExchange CDC Publisher from a test environment to a production environment but retain the Avro-related properties.

Important: Do not include the same property in multiple configuration files. If the same property appears with different values in multiple configuration files, the PowerExchange CDC Publisher will issue an error.
Common Configuration Properties

The sample cdcPublisherCommon.cfg file contains properties that identify the PowerExchange CDC Publisher host and port.

Property Descriptions

**Common.pwxCDCPublisherHost**

The host name or IP address of the machine where the PowerExchange CDC Publisher server runs. Enter `localhost` if you plan to run the PwxCDCAdmin and PwxCDCInfo utilities on the same machine as the PowerExchange CDC Publisher server.

Tip: If you plan to run the PwxCDCAdmin or PwxCDCInfo utilities on a system that is remote from the PowerExchange CDC Publisher server, enter a public IP address or host name.

Default value is `localhost`.

**Common.pwxCDCPublisherPort**

Required. The port number that the PowerExchange CDC Publisher uses on the specified host to listen for command and control requests.

No default value is provided.

**Common.logPwxConfigContents**

Indicates whether to write CDCPUB_10067 messages that report the contents of the PowerExchange dbmover configuration file on the CDC Publisher system to the message log. The default value of true causes these messages to be written to the log. If any dbmover statements contain sensitive information such as passwords, you can set this property to false to prevent the messages from being logged.

Default value is `true`.

Avro Formatter Configuration Properties

The sample cdcPublisherAvro.cfg file contains configuration properties that define the format of the generated Avro schema and the encoding type to use for serializing the Avro records to be included in messages. You must set the schema format and encoding type.

Property Descriptions

The following properties are in the sample cdcPublisherAvro.cfg file:

**Formatter.formatterType**

The type of data serialization formatter to use for messages. The only valid value is `Avro`.

**Formatter.avroSchemaFormat**

Required. The Avro schema format that the PowerExchange CDC Publisher uses to generate the Avro schema that will determine the structure of the message values. Valid values are:

- `avroFlatSchemaFormatV1`. Structures messages by using a flat Avro schema format, which lists all Avro fields in one Avro record. A unique Avro schema is generated for each source object, which contains the Avro field definitions.

- `avroNestedSchemaFormatV1`. Structures messages by using a nested Avro schema format, which provides a main Avro record that contains a separate nested record for each type of Avro field.
• **avroGenericSchemaFormatV1.** Structures messages in a generic manner that accommodates any source object definition. All source columns are represented by an array. Each array entry contains column data and metadata. The source column names are included in each data record, allowing the generic schema to be independent of the source table.

No default value is provided.

**Note:** You can "wrap" a flat, nested, or generic schema by setting the Formatter.avroWrapperSchemaFormat property to avroWrapperSchemaFormatV1. The schema then consists of four fields for each source object.

Use a generic or wrapper schema to allow a single Avro schema to represent multiple source tables. For more information about the schema formats, see Appendix B, "Avro Schema Formats" on page 69.

**Formatter.avroEncodingType**

Required. The Avro encoding type that the CDC Publisher Formatter uses to serialize the Avro records to be included in messages. Valid values are:

- **binary.** Use binary encoding to serialize Avro records.
- **json.** Use JSON to serialize Avro records.
- **none.** Do not use any explicit encoding type. Specify this option only if you use Confluent Schema Registry to store and retrieve Avro schemas.

No default value is provided.

The following additional properties can also be included in the cdcPublisherAvro.cfg file at your discretion:

**Formatter.avroAsString**

Controls whether change data with a binary datatype is represented as string data in Avro messages. Set this property to true if the data will be consumed by applications that do not support binary data, such as Informatica Intelligent Streaming. The default value is false.

**Formatter.avroIncludeBeforeImage**

Controls whether the generated Avro schema and messages include a field for before-image data. Set this property to true to include this field. Set this property to false to not include this field.

If you include the before-image field, the field is populated with data for UPDATE operations, if you set the Extract.pwxUpdateImageOption property to enable the extraction of before-image data from the PowerExchange change stream. For DELETE and INSERT operations, the field is not populated with data.

The default value is true.

**Formatter.avroBinaryStringRepresentationType**

If you set the Formatter.avroBinaryAsString property to true or use a generic Avro format, indicates whether binary data is represented as a hexadecimal string or base64 string. Valid values are:

- **hexadecimal**
- **base64**

The default value is base64.

**Formatter.avroSchemaPrintDefaultFields**

Controls whether Avro schemas include the "default" fields. If you need to reduce the schema size, you can set this property to false to exclude the default fields. The default value is true, which includes the default fields.
Formatter.avroSchemaPrintDocFields

Controls whether Avro schemas include the "doc" fields. The doc fields include metadata such as the CDC and PowerExchange datatypes, precision, and scale. If you need to reduce the schema size, you can set this property to false to exclude the doc fields. The default value is true, which includes the doc fields.

Formatter.avroSchemaPrintPretty

Controls whether Avro schemas include spaces and line feeds to improve legibility. If you need to reduce the schema size, you can set this property to false to exclude the spaces and lines feeds. The default is true, which includes the spaces and line feeds.

Formatter.avroWrapperSchemaFormat

Enables the use of an Avro "wrapper" schema format. The wrapper schema can be used to describe any source object. The wrapper, or parent, schema consists of four fields for each source object: the sequence number of the change record, source table name, change operation type, and the "wrapped" Avro child schema expressed as a large string. The consumer application can then parse the underlying data and put it in the proper Avro format for the source object. To use a wrapper schema format, set this property to avroWrapperSchemaFormatV1. No default value is provided. For more information, see "Avro Wrapper Schema Format" on page 71.

Formatter.avroUseLogicalDateType

Formatter.avroUseLogicalDecimalType

Formatter.avroUseLogicalTimeMillisType

Formatter.avroUseLogicalTimeMicrosType

Formatter.avroUseLogicalTimestampMillisType

Formatter.avroUseLogicalTimestampMicrosType

If you use Avro logical types for dates, decimal values, times, or timestamps and want the CDC Publisher to make a best-effort attempt to process these logical types, set this property to true. The following sets of properties are mutually exclusive so specify one property or the other but not both:

- Formatter.avroUseLogicalTimeMillisType and Formatter.avroUseLogicalTimeMicrosType
- Formatter.avroUseLogicalTimestampMillisType and Formatter.avroUseLogicalTimestampMicrosType

The default value for each of these properties is false.

Note: If you set a property to true, make sure that the source fields are defined in the extraction map with a compatible data type, scale, and precision.

Kafka Connector Configuration Properties

The sample cdcPublisherKafka.cfg file contains configuration properties related to the Apache Kafka target.

Property Descriptions

The following properties are in the sample cdcPublisherKafka.cfg configuration file:

Connector.queueType

The type of target messaging queue to which PowerExchange CDC Publisher streams change data. The only valid value is Kafka.

Connector.kafkaTopic

Required. The Kafka topic or topics to which PowerExchange CDC Publisher sends messages that contain extracted change data. Enter a specific Kafka topic name if you want to send all messages to a
single topic that you created, or enter USE_TABLE_NAME to direct Kafka to generate a separate topic for each source table using the mapname_tablename portion of the full extraction map name as the topic name. The full extraction map name has the format scheme.mapname_tablename.

Note: This parameter is optional. By default, the auto.create.topics.enable parameter in the Kafka server.properties file is set to true. With this setting, the Kafka broker automatically generates the topic or topics if they do not exist the first time messages are sent to the topics. If you set the auto.create.topics.enable parameter to false, Kafka does not generate the topics. You must set this property to determine how the topics will be created.

No default value is provided.

**Connector.kafkaProducerPropertiesFile**

Required. The path and file name of the Kafka producer.properties file that PowerExchange CDC Publisher uses to communicate with Kafka. This file is typically in the /Kafka_installation/config directory.

No default value is provided.

The following additional properties can also be included in the cdcPublisherKafka.cfg file at your discretion:

**Connector.kafkaCheckpointFileDirectory**

A directory name for the instance subdirectory to which the CDC Publisher writes checkpoint files. The default value is "checkpoint." You can use this property to override the default directory name.

**Connector.kafkaFileCheckpointFileName**

Controls the name of the file to which the CDC Publisher writes checkpoints. This file is in the directory that is specified by the Connector.kafkaCheckpointFileDirectory property. Enter a specific file name or DEFAULT. If you enter DEFAULT, the CDC Publisher uses the default file name of "checkpoint." The default property value is DEFAULT.

**Connector.kafkaProducerGuaranteeDelivery**

Controls whether the CDC Publisher uses guaranteed delivery to write a checkpoint after each change operation is acknowledged as successfully received by Kafka. This delivery mode avoids message loss and duplication but slows apply processing. Set to property to false if you do not want to use guaranteed delivery. With the setting of false, if Kafka terminates while changes are in flight, the CDC Publisher might apply duplicate messages to the target messaging system after the change stream is restarted. The default value is true.

### PowerExchange Extract Configuration Properties

The sample cdcPublisherPowerExchange.cfg file contains properties that the PowerExchange CDC Publisher uses to extract change from PowerExchange and to retrieve PowerExchange extraction map definitions of source objects. The following list describes the properties in the sample file. It also includes some additional properties that you might want to use.

**Property Descriptions**

The following properties are in the sample cdcPublisherPowerExchange.cfg file:

**Extract.pwxCapiConnectionName**

Required. The NAME parameter value in the CAPX CAPI_CONNECTION statement in the PowerExchange DBMOVER configuration file. This CAPI_CONNECTION statement is used to connect to the PowerExchange change data stream to read data from PowerExchange Logger log files.

No default value is provided.
Extract.pwxExtractionMapSchemaName

Required. The PowerExchange-generated schema name that is included in the names of the PowerExchange extraction maps for the source objects. This schema name has the format uninstance or dnninstance, where “u” indicates a user-defined extraction map, “d” indicates the default extraction map generated from the capture registration, nn is a two-digit value that represents the source type, and instance is the instance value from the PowerExchange registration group for the data source. The schema name is the first part of the full extraction map name: xnninstance.mapname_tablename. This property is used to select the extraction maps that the CDC Publisher will use to extract change data.

Note: All extraction maps with names that match the specified schema name or mask are selected. If you want to use a subset of these extraction maps, define the Extract.captureConsumerExcludeNameList or Extract.captureConsumerIncludeNameList property to filter the extraction maps.

No default value is provided.

Extract.captureConsumerIncludeNameList

A list of the extraction maps that match the specified schema name and that you want to explicitly include in extraction processing. The PowerExchange CDC Publisher process will use only these included extraction maps. Any other extraction maps that were selected based on the extraction-map schema name are ignored. For each extraction map to include, enter the full extraction map name in the format xnninstance.mapname_tablename. Use a comma (,) separator between the values, for example:

Extract.captureConsumerIncludeNameList=xnninstance.mapname_tablenamel,xnninstance.mapname_tablename2, ...

You can include the asterisk (*) wildcard anywhere in the value to represent one or more characters, thereby creating a mask that matches multiple extraction maps.

Extract.captureConsumerExcludeNameList

A list of the extraction maps that match the specified schema name but that you want to exclude from extraction processing. For each extraction map to exclude, enter the full extraction map name in the format xnninstance.mapname_tablename. Use a comma (,) separator between the values, for example:

Extract.captureConsumerExcludeNameList=xnninstance.mapname_tablenamel,xnninstance.mapname_tablename2, ...

You can include the asterisk (*) wildcard anywhere in the value to represent one or more characters, thereby creating a mask that matches multiple extraction maps.

Extract.pwxNodeLocation

Required. The node name (first positional parameter) in the NODE statement that is used to connect to the PowerExchange Listener for change data extraction. This statement is defined in the PowerExchange DBMOVER configuration file. This Listener runs on the system with the PowerExchange Logger for Linux, UNIX, and Windows log files.

No default value is provided.

Extract.pwxNodeUserId

A user ID that is authorized to access to the node that contains the PowerExchange Logger log files from which change data will be extracted. Define this property only if access to the node specified in the Extract.pwxNodeLocation property is secured by a user ID and password. If you use remote logging of source data to PowerExchange Logger for Linux, UNIX, and Windows log files that reside off the source system, this property value matches the CAPTURE_NODE_UID parameter value in the PowerExchange Logger configuration file, pwxccl.cfg.

No default value is provided.
Extract.pwxNodePwd

A password that is associated with the user ID specified in the Extract.pwxNodeUserId property. This password can be a clear-text password or a password that has been encrypted by running the PwxCDCAdmin utility with the ENCRYPT parameter.

Define this property only if access to the node specified in the Extract.pwxNodeLocation property is secured by a user ID and password. If you use remote logging of source data to PowerExchange Logger for Linux, UNIX, and Windows log files that reside off the source system, this property value matches the CAPTURE_NODE_PWD parameter value in the PowerExchange Logger configuration file, pwxccl.cfg.

If you use PowerExchange-encrypted passwords, specify the Extract.pwxNodeEPwd property instead.

No default value is provided.

Extract.pwxNodeEPwd

An encrypted password that is associated with the user ID specified in the Extract.pwxNodeUserId property. This password must have been encrypted by using the PowerExchange Navigator Encrypt Password facility.

Define this property only if access to the node specified in the Extract.pwxNodeLocation property is secured by a user ID and password. If you use remote logging of source data to PowerExchange Logger for Linux, UNIX, and Windows log files that reside off the source system, this property value matches the CAPTURE_NODE_EPWD parameter value in the PowerExchange Logger configuration file, pwxccl.cfg.

If you use clear-text passwords or passwords that have been encrypted by the CDC Publisher PwxCDCAdmin ENCRYPT command, specify the Extract.pwxNodePwd property instead.

No default value is provided.

Extract.pwxXmapLocation

The node name (first positional parameter) in the NODE statement that is used to connect to the PowerExchange Listener on the source system to retrieve extraction maps for source objects. This statement is defined in the local PowerExchange DBMOVER configuration file. If you use remote logging of source data to PowerExchange Logger for Linux, UNIX, and Windows log files that reside off the source system, this property value matches the CAPTURE_NODE parameter value in the PowerExchange Logger configuration file, pwxccl.cfg.

Note: If the data source and extraction maps are on the same node as the PowerExchange Listener and Logger, you do not need to specify this property. You can specify only the Extract.pwxNodeLocation property to connect the PowerExchange Listener that handles both requests for extraction maps and requests for change data extraction.

No default value is provided.

Extract.pwxXmapUserId

A user ID that is authorized to access the node that contains the PowerExchange extraction maps for source objects. Define this property only if access to the node specified in the Extract.pwxXmapLocation property is secured by a user ID and password. If you use remote logging of source data to PowerExchange Logger for Linux, UNIX, and Windows log files that reside off the source system, this property value matches the CAPTURE_NODE_UID parameter value in the PowerExchange Logger configuration file, pwxccl.cfg.

No default value is provided.

Extract.pwxXmapPwd

A clear-text or encrypted password that is associated with the user ID specified in the Extract.pwxXmapUserId property. To encrypt the password, run the PwxCDCAdmin utility with the EXTRACT parameter.
Define this property only if access to the node specified in the Extract.pwxXmapLocation property is secured by a user ID and password. If you use remote logging of source data to PowerExchange Logger for Linux, UNIX, and Windows log files that reside off the source system, this property value matches the CAPTURE_NODE_PWD parameter value in the PowerExchange Logger configuration file, pwxcl.cfg.

No default value is provided.

The following additional properties can also be included the cdcPublisherPowerExchange.cfg file at your discretion:

**Extract.pwxReturnLogInfoMessages**

Controls whether the informational messages that PowerExchange returns to the CDC Publisher are written to the PwxCdcPublisher.out file in the logs subdirectory. Set this property to true to log the PowerExchange informational messages, or set this property to false to not log these informational messages. Default is false.

**Extract.pwxUpdateImageOption**

For UPDATE operations, indicates whether the PowerExchange CDC Publisher receives both the before-image and after-image data or only the after-image data from the PowerExchange change stream. Valid values are:

- **AI.** Gets the after image only.
- **TU.** Gets the before image followed by the after image.
- **AB.** Gets the after image followed by the before image.

The default value is TU. To a CDC Publisher user, the AB and TU options produce the same results.

**Important:** If the PowerExchange Logger for Linux, UNIX, and Windows is configured to log only after images, you must set this property to AI. Otherwise, PowerExchange issues a fatal error.

**Extract.restart1**

The PowerExchange sequence token value that you want to use to restart the CDC Publisher process from a specific point in the change stream. For this property value to be used, you must restart the PwxCdcPublisher process with the COLDSTART=Y or RESTART=FROM_CONFIG parameter. You must also specify the Extract.restart2 property. No default value is provided.

**Extract.restart2**

The PowerExchange restart token value that you want to use to restart the CDC Publisher process from a specific point in the change stream. For this property value to be used, you must restart the PwxCdcPublisher process with the COLDSTART=Y or RESTART=FROM_CONFIG parameter. You must also specify the Extract.restart1 property. No default value is provided.

---

Filtering Extraction Maps and Source Objects

PowerExchange CDC Publisher uses the pwxExtractionMapSchemaName property value to select the PowerExchange extraction maps to use for extracting change data from source objects. You can filter this
To filter extraction maps, enter one or both of the following parameters in the cdcPublisherPowerExchange.cfg configuration file:

- **Extract.captureConsumerIncludeNameList.** Specifies a subset of the selected extraction maps to explicitly include in extraction processing. For each extraction map to include, enter the full extraction map name in the format `xnninstance.mapname_tablename`. Use a comma (,) separator between the values. For example:

  ```
  Extract.captureConsumerIncludeNameList=xnninstance.mapname_tablenamel,xnninstance.mapname_tablename2,...
  ```

  You can use the asterisk (*) wildcard anywhere in the value to represent one or more characters, thereby creating a mask that matches multiple extraction maps. The PowerExchange CDC Publisher process will use the included extraction maps and ignore any other extraction maps that matched the extraction-map schema name.

  Use this parameter when you want to explicitly include some of the selected extraction maps and ignore all of the other extraction maps that matched the `pwxExtractionMapSchemaName` value.

- **Extract.captureConsumerExcludeNameList.** Specifies a subset of the selected extraction maps to exclude from extraction processing. For each extraction map to exclude, enter the full extraction map name in the format `xnninstance.mapname_tablename`. Use a comma (,) separator between the values. You can include the asterisk (*) wildcard anywhere in the value to represent one or more characters, thereby creating a mask that matches multiple extraction maps.

  Use this parameter when you want to use most of the extractions maps that matched the `pwxExtractionMapSchemaName` value, excluding only those that match the exclusion criteria in this property.

You can use the `Extract.captureConsumerIncludeNameList` and `Extract.captureConsumerExcludeNameList` properties together. For example, use an `Extract.captureConsumerIncludeNameList` mask to select a subset of extraction maps and then use the `Extract.captureConsumerExcludeNameList` to narrow the extraction maps in that subset.

For example, assume the following properties are specified:

- **Extract.pwxExtractionMapSchemaName=d8district1**
- **Extract.captureConsumerIncludeNameList=d8district1.store***
- **Extract.captureConsumerExcludeNameList=d8district1.store1_sales, d8district1.store1_customers, d8district1.store1_items**

In this case, all extraction maps that have names beginning with the `xnninstance` (schema) value of "d8district1" are selected, where the full extraction map name format is `xnninstance.mapname_tablename`. Of the selected extraction maps, you explicitly include only those that have `mapname` values that begin with "store." From the subset of included extraction maps, you explicitly exclude the three extraction maps for map "store 1."

**Note:** The PowerExchange CDC Publisher does not automatically exclude extraction maps that are associated with capture registrations that have the status of Inactive, History or Suspended. If no change data is available, the CDC Publisher does not publish change data for the registered source objects. If you will never use these deactivated capture registrations or their associated extraction maps to capture and publish change data, you can exclude the extraction maps in the CDC Publisher to reduce resource consumption. Before you define the exclusion filter, ensure that any change data that was previously captured for the source object has been published to the target messaging system. After you define the exclusion filter, you must restart the CDC Publisher process for the filter to take effect.
Configuring CDC Publisher Message Logging

The PowerExchange CDC Publisher uses the Apache Log4j 2 API to implement message logging and tracing. The PWX CDC Publisher provides a sample Log4j logging configuration file in XML format, PwxCDCP publicationLog4j.xml, in the \pwxcdcpub_root\installation\samples subdirectory. The sample file contains the Log4j 2 configuration properties that the PowerExchange CDC Publisher uses.

**Note:** For information about Log4j 2 logging configuration properties, see the Apache Log4j 2 documentation.

You must copy the sample PwxCDCPpublicationLog4j.xml file from the samples subdirectory to the \pwxcdcpub_root\installation\instanceA\config subdirectory. If you use multiple instances, copy the file to the config subdirectory for each instance. Do not rename the file. Usually, the PwxCDCPpublicationLog4j.xml file in the config subdirectory can be used without modification, even for instances other than instanceA. However, if you want trace messages to be logged, you must change the following Logger statement to specify level="all":

```xml
<Logger name="trace" level="all" additivity="false" includeLocation="false">
  <AppenderRef ref="TraceFile"/>
</Logger>
```

Based on the PwxCDCPpublicationLog4j.xml file configuration, as delivered, Log4j 2 logs CDC Publisher messages to files in the \pwxcdcpub_root\installation\instanceA\logs subdirectory. If you use instances other than instanceA, the PwxCDCPpublicationLog4j.xml file is automatically updated with the current instance name when you start the PwxCDCPpublication process so that messages will be logged to the logs subdirectory of that instance.

Also, based on the delivered PwxCDCPpublicationLog.xml file, Log4j 2 generates log files for PowerExchange CDC Publisher messages in the \pwxcdcpub_root\installation\instance\logs subdirectory using the following naming conventions:

- The active log file is named PwxCdcPublisher.out.
- Archive log files are generated using the following file-name format:
  PwxCdcPublisher.out.yyyy-mm-dd-hh-n.gz, where yyyy-mm-dd is year-month-day, hh is hour of the day, and n is a generated number. In the a sample Log4j logging configuration file, the maximum number of archive log files is set to 999999.

The active log is archived when the maximum file size is reached or when the CDC Publisher is restarted. The Log4j default maximum size is 10 MB. You can change this maximum size in the PwxCDCPpublicationLog4j.xml file. Each time you restart the CDC Publisher, the previous log file is archived and a new active log file is created.

For your convenience, any error and warning messages that PowerExchange returns to the CDC Publisher are also written to the PwxCdcPublisher.out file in the logs subdirectory by default. Optionally, you can configure the CDC Publisher to also log PowerExchange informational messages to the PwxCdcPublisher.out file by specifying Extract.pwxReturnLogInfoMessages=true in the cdcPublisherPowerExchange.cfg configuration file.

If Informatica Intelligent Streaming Will Consume Data from Kafka

PowerExchange CDC Publisher streams change data that PowerExchange captures from heterogenous sources in near real time to Apache Kafka messaging systems. Informatica Intelligent Streaming can then
consume the change data from the Kafka message queue for a variety of purposes, such as generating near-
real-time fraud detection alerts or customized sales offers at point-of-sale.

If the Informatica Intelligent Streaming product will consume the change data that the PowerExchange CDC
Publisher sends to an Apache Kafka target, use the following PowerExchange CDC Publisher configuration
guidelines:

- Informatica Intelligent Streaming cannot consume data from fields that have a binary datatype. Configure
  the PowerExchange CDC Publisher to send data from binary fields as string data by setting the following
  properties in the cdcPublisherAvro.cfg configuration file:

  - **Formatter.avroBinaryAsString=true.** With this setting, binary data is represented as string data in the
    generated Avro messages.

  - **Formatter.avroBinaryStringRepresentationType=(base64|hexadecimal).** When
    Formatter.avroBinaryAsString=true, this property determines whether to use base64 or hexadecimal
    strings to represent binary data. Default is base64.

- Informatica Intelligent Streaming cannot consume JSON-encoded Avro messages. To use binary-encoded
  messages, specify **Formatter.avroEncodingType=binary** in the cdcPublisherAvro.cfg configuration file.

- As a consumer application, Informatica Intelligent Streaming must have copies of the Avro schemas for
  the source tables to properly interpret the change data in the Kafka messages. You can use the
  REPORT=FORMAT parameter of the Pwx CDCAdmin utility to report the existing Avro schemas in a legible
  format for use by consumer applications. If no Avro schemas have been generated for the source tables,
  the utility attempts to create the Avro schemas based on the properties in the cdcPublisherAvro.cfg
  configuration file. For more information, see "PwxCDCAdmin Utility - Command and Parameters" on page
  66.

- If you try to import an Avro schema that the PowerExchange CDC Publisher generated for a very large
  table and that is larger than 65535 bytes into Informatica Intelligent Streaming, the Scala compiler issues
  a Java exception related to the scala.tools.asm package. This problem occurs because the Scala code
does not handle literals greater than 65535 bytes in size. To circumvent this problem, you can configure
  the PowerExchange CDC Publisher to generate Avro schema in a minimized format by specifying some or
  all of the following properties in the cdcPublisherAvro.cfg configuration file:

  - **Formatter.avroSchemaPrintPretty={true|false}.** Set this property to false to not include the spaces and
    line feeds that are intended to improve legibility in the generated Avro schemas. Default value is true,
    which causes the spaces and line feeds to be included.

  - **Formatter.avroSchemaPrintDocFields={true|false}.** Set this property to false to not report the "doc"
    fields in the generated Avro schemas. The doc fields include metadata such as the CDC and
    PowerExchange datatypes, precision, and scale. Default value is true, which causes this information to
    be included.

  - **Formatter.avroSchemaPrintDefaultFields={true|false}.** Set this property to false to not include the
    "default" fields in the generated Avro schemas. Default value is true, which causes the default fields to
    be included.
CHAPTER 7

Streaming Change Data

This chapter includes the following topics:

- Before You Start Data Streaming, 48
- Starting the PwxCDCPublisher Utility, 48
- Operational Considerations, 49

Before You Start Data Streaming

To begin streaming change data from a PowerExchange source to a target messaging system, you start the main PwxCDCPublisher utility.

The PwxCDCPublisher utility can run as a Linux daemon in the foreground or background or as a Windows process in the foreground.

Before you start the PwxCDCPublisher utility, ensure that the following prerequisites are met:

- Active capture registrations and extraction maps exist for the data source objects.
- The PowerExchange Listener and PowerExchange Logger for Linux, UNIX, and Windows are running.
- PowerExchange has captured change records from the data sources and logged them in PowerExchange Logger log files.
- Apache Kafka is running.

Starting the PwxCDCPublisher Utility

You can start the PwxCDCPublisher utility from a command prompt at the pwxcdcpub_root_installation\bin subdirectory or by using batch file, provided that file system security allows you to run this script. This procedure describes how to start the utility from a command prompt.

Before you start the PwxCDCPublisher, ensure that Apache Kafka and Zookeeper are running. Also ensure that the PowerExchange Listener instances in the PowerExchange environment are running.

> At the command prompt, enter the PwxCDCPublisher command. Include the optional INSTANCE parameter if you want to run the PwxCDCPublisher process using an instance other than the default "instanceA" at /pwxcdcpub_root_installation/instanceA.
Operational Considerations

The following operational considerations apply to PowerExchange CDC Publisher change data streaming:

- You can configure multiple PowerExchange CDC Publisher "instances," each with its own set of checkpoint files, configuration files, log files, and Avro schemas. When you start a PwxCDCPublisher process, you specify the name of the instance to use. The instance name corresponds to a directory within the PowerExchange CDC Publisher installation. By default, the PowerExchange CDC Publisher installation creates the "instanceA" directory and its subdirectories. Consider adding instances to process change data in multiple streams for different sources or different targets or for load balancing.

- The PowerExchange CDC Publisher reads extraction map information only at startup. If you edit the capture registrations and the associated extraction maps for the data sources, PowerExchange CDC Publisher does not automatically update the Avro schema. You must run the PwxCDCAdmin utility with the CLEAR parameter to remove the old Avro schema from cache. You can then run the utility again with the RESET parameter to re-read the extraction maps and cache the updated Avro schema. After you restart the PwxCDCPublisher process, it can use the updated Avro schema in cache to generate messages in the latest schema format for transmittal to the target messaging system. For more information about PwxCDCAdmin utility parameters, see "PwxCDCAdmin Utility - Command and Parameters" on page 66.

- If you start the PwxCDCPublisher process before the PowerExchange Logger for Linux, UNIX, and Windows has written at least one source change record to its log files, the Publisher Extractor process might end with an error. This issue is caused by a PowerExchange Logger reader limitation that pertains to the first run of the PowerExchange Logger.
• PowerExchange sends change data from successful units of work (UOWs) to the CDC Publisher in commit order. The PowerExchange CDC Publisher maintains this order when streaming messages to the target messaging system.

• The CDC Publisher processes a stream of change records in a manner that is independent from transaction start and end points. If the CDC Publisher terminates or connectivity to the target messaging system is lost before all of the changes in a transaction are processed, the target will have only some of the changes until you restart the CDC Publisher so that it can send the remaining changes in the transaction. The CDC Publisher process will restart from the last checkpoint position recorded in the checkpoint file by default.

• The PowerExchange CDC Publisher can filter source tables or objects by using the filtering criteria you specify in the Extract.captureConsumerIncludeNameList and Extract.captureConsumerExcludeNameList properties in the cdcPublisherPowerExchange.cfg configuration file. However, the CDC Publisher cannot filter source columns.

• The PowerExchange CDC Publisher does not automatically exclude extraction maps that are associated with capture registrations that have the status of Inactive, History or Suspended. If no change data is available, the CDC Publisher does not publish change data for the registered source objects. If you will never use these deactivated capture registrations or their associated extraction maps to capture and publish change data, you can exclude the extraction maps in the CDC Publisher to reduce resource consumption. Before you define the exclusion filter, ensure that any change data that was previously captured for the source object has been published to the target messaging system. After you define the exclusion filter, you must restart the CDC Publisher process for the filter to take effect. If you plan to reactivate a suspended registration, first remove the exclusion filter for the associated extraction maps. Then restart the CDC Publisher process and reactivate the registration.
CHAPTER 8

Monitoring PowerExchange CDC Publisher

This chapter includes the following topics:

- Monitoring Overview, 51
- Determining the Status of CDC Publisher Processes, 51
- Generating Statistics on CDC Publisher Processing, 52

Monitoring Overview

You can use the PwxCDCInfo utility to report PowerExchange CDC Publisher status information and internal statistics for CDC Publisher processes, updated target topics, and CDC Publisher internal attributes.

The PwxCDCInfo utility can report the following types of monitoring information:

- The overall status of the PowerExchange CDC Publisher process and the status of each major subprocess (Extractor, Formatter, and Connector). The status is designated by a color indicator: "GREEN" for running, "YELLOW" for a warning situation, and "RED" for stopped.
- Information about the target topics that the CDC Publisher updated, including the topic name, the associated source table name, the number of messages sent to the topic, and the last change operation type and sequence ID.
- Statistics for diagnostic use, by internal attribute and CDC Publisher child process.

Also, the PowerExchange CDC Publisher messages that are written to the logs subdirectory provide information about errors, potential errors, and processing status. For more information, see Appendix C, "Message Reference" on page 73.

Determining the Status of CDC Publisher Processes

To determine the overall status of PowerExchange CDC Publisher processes, run the PwxCDCInfo utility with the STATUS parameter. To provide status information for each CDC Publisher process (Extractor, Formatter, and Connector), include the optional DETAIL parameter.

You can use this information to determine if a CDC Publisher process stopped or has a potential problem.
Before you issue the PwxCDCInfo command, ensure that the PowerExchange CDC Publisher process is running in a separate command line window.

- At the command prompt, enter the PwxCDCInfo command with the STATUS parameter. Include the optional INSTANCE parameter if the Publisher process is using an instance other than the default "instanceA" at /pwxpub_root installation/instanceA. 

  Use the following syntax:

  `PwxCDCInfo STATUS [DETAIL] [VERBOSE] [INSTANCE=override_instance_directory]`

  You can issue the command from the pwxcdcpub_root installation\bin subdirectory or from another location if you specified the PWXPUB_HOME environment variable.

  For the full command syntax, see "PwxCDCInfo Utility - Command and Parameters" on page 64.

  The command output is displayed on screen. Optionally, you can direct the output to a file.

  - If you entered PwxCDCInfo STATUS, the output shows the overall status of all PowerExchange CDC Publisher processes by using a color indicator and brief description. For example:

    `PROCESS STATE  GREEN  All processes are running`

    The other process states are:

    - YELLOW. A warning condition exists. The data stream might be restricted or not flowing.
    - RED. The CDC Publisher processes are not running or cannot be found.

  - If you entered PwxCDCInfo STATUS DETAIL, the output shows the status color indicator for each CDC Publisher child process. For example:

    `PARENT   CHILD   ATTRIBUTE  VALUE  DESCRIPTION`

    `Extract_Connector Extract  TRAFFIC LIGHT GREEN  
    Extract_Connector Formatter  TRAFFIC LIGHT GREEN  
    Extract_Connector Connector  TRAFFIC LIGHT GREEN  

  - If you entered PwxCDCInfo STATUS DETAIL VERBOSE, the output shows the status color indicator for each CDC Publisher child process with a description of the valid status values. For example:

    `PARENT   CHILD   ATTRIBUTE  VALUE  DESCRIPTION`

    `Extract_Connector Extract  TRAFFIC LIGHT GREEN  Similar to a traffic light, a red value 
    indicates an error, 
    a green value indicates the process is running as expected, and a yellow value indicates a mixed 
    state (not an 
    error and not running as expected).`

    `Extract_Connector Formatter  TRAFFIC LIGHT GREEN  Similar to a traffic light, a red value 
    indicates an error, 
    a green value indicates the process is running as expected, and a yellow value indicates a mixed 
    state (not an 
    error and not running as expected).`

    `Extract_Connector Connector  TRAFFIC LIGHT GREEN  Similar to a traffic light, a red value 
    indicates an error, 
    a green value indicates the process is running as expected, and a yellow value indicates a mixed 
    state (not an 
    error and not running as expected).`

**Generating Statistics on CDC Publisher Processing**

You can use the PwxCDCInfo utility with the STATS parameter to generate statistics and state information for the topics to which the CDC Publisher sent messages and for CDC Publisher internal attributes for diagnostic use.

The statistics are cumulative from the start time of the PowerExchange CDC Publisher process. You can reset the internal statistics to their original values to determine the processing activity from the reset point going forward.
Before you issue the PwxCDCInfo command, ensure that the PowerExchange CDC Publisher process is running in a separate command line window.

- At the command prompt, enter the PwxCDCInfo command with the STATS=TOPIC or STATS=INTERNAL parameter. Include the optional INSTANCE parameter if the Publisher process is using an instance other than the default "instanceA" at /pwxcdcpub_root_installation/instanceA.

Use the following syntax:

```
PwxCDCInfo STATS=[TOPIC|INTERNAL] [DETAIL [VERBOSE]] [RESET]
[INSTANCE=override_instance_directory]
```

You can issue the command from the pwxcdcpub_root_installation\bin subdirectory or from another location if you specified the PWXPUB_HOME environment variable.

Tip: For the full command syntax, see "PwxCDCInfo Utility - Command and Parameters" on page 64.

The command output is displayed on screen. Optionally, you can direct the output to a file.

If you entered STATS=TOPIC, the output shows processing information for each topic that the CDC Publisher updated, including a count of messages sent to the topic and the last change operation type and sequence ID. The following example shows information for one updated topic:

```
Topic updated: abcd0006_SRC_ALLCHAR_16K
PartitionID of the topic: 0
Count of messages sent: 8
Partition offset returned: 7
Message key:
Source table name: dRcollnam1,abcd0006_SRC_ALLCHAR_16K
Type of last change event: INSERT EVENT
Sequence ID of last change:2,PWX GENERIC,1,2,3,D4002CA219131A00000000000002CA2191319000001F000037
C30000170012400100000010000000000,00,000001BFDF3288319
```

Note: The CDC Publisher generates the Sequence ID value and uses it internally to track the last change applied to the topic.

You can use the STATS=TOPIC RESET command to clear the statistics.

If you entered STATS=INTERNAL, the output shows internal statistics and state information, by attribute, for each type of PowerExchange CDC Publisher child process. This information is primarily for diagnostic use by Informatica Global Customer Support and other technical staff. The following example shows a small portion of this report:

```
<table>
<thead>
<tr>
<th>PARENT</th>
<th>CHILD</th>
<th>ATTRIBUTE</th>
<th>VALUE</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>Extract</td>
<td>Extract Capture Consumer</td>
<td>ErrorState</td>
<td>false</td>
<td>is True if the process is currently in error, false otherwise.</td>
</tr>
<tr>
<td>Extract</td>
<td>Extract Capture Consumer</td>
<td>IsRunningState</td>
<td>true</td>
<td>True if the process is running, false if it has stopped running.</td>
</tr>
<tr>
<td>Extract</td>
<td>Extract Capture Consumer</td>
<td>IsSleepingState</td>
<td>true</td>
<td>True if the process is sleeping, false if not.</td>
</tr>
<tr>
<td>Extract</td>
<td>Extract Capture Consumer</td>
<td>SleepDuration</td>
<td>2879102</td>
<td>If sleeping, the current duration of that sleep (milliseconds).</td>
</tr>
<tr>
<td>Extract</td>
<td>Extract Capture Consumer</td>
<td>InputRecordsRead</td>
<td>4761</td>
<td>The current number of input records consumed by this process.</td>
</tr>
<tr>
<td>Extract</td>
<td>Extract Capture Consumer</td>
<td>InputBytesRead</td>
<td>10976086</td>
<td>The current number of bytes of input consumed by this process.</td>
</tr>
</tbody>
</table>
```

Tip: The STATS=INTERNAL output is displayed in a wide format. For easier reading, direct the output to a file.

You can use STATS=INTERNAL RESET command to set the statistics back to their original values.

If you entered STATS=INTERNAL VERBOSE, the output shows the same type of diagnostic information as the STATS=INTERNAL report with an additional description of each attribute. The following example shows a small portion of this report:

```
<table>
<thead>
<tr>
<th>PARENT</th>
<th>CHILD</th>
<th>ATTRIBUTE</th>
<th>VALUE</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>Extract</td>
<td>Extract Capture Consumer</td>
<td>ErrorState</td>
<td>false</td>
<td>is True if the process is currently in error, false otherwise.</td>
</tr>
<tr>
<td>Extract</td>
<td>Extract Capture Consumer</td>
<td>IsRunningState</td>
<td>true</td>
<td>True if the process is running, false if it has stopped running.</td>
</tr>
<tr>
<td>Extract</td>
<td>Extract Capture Consumer</td>
<td>IsSleepingState</td>
<td>true</td>
<td>True if the process is sleeping, false if not.</td>
</tr>
<tr>
<td>Extract</td>
<td>Extract Capture Consumer</td>
<td>SleepDuration</td>
<td>2596322</td>
<td>If sleeping, the current duration of that sleep (milliseconds).</td>
</tr>
<tr>
<td>Extract</td>
<td>Extract Capture Consumer</td>
<td>InputRecordsRead</td>
<td>4761</td>
<td>The current number of input records consumed by this process.</td>
</tr>
<tr>
<td>Extract</td>
<td>Extract Capture Consumer</td>
<td>InputBytesRead</td>
<td>10976086</td>
<td>The current number of bytes of input consumed by this process.</td>
</tr>
</tbody>
</table>
```

Generating Statistics on CDC Publisher Processing 53
<table>
<thead>
<tr>
<th>Extract</th>
<th>Extract Capture Consumer</th>
<th>LastSequenceIn</th>
<th>2, FWK_GENERIC, 1, 2, ...</th>
<th>The sequence value of the last operation read into this process.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Extract</td>
<td>Extract Capture Consumer</td>
<td>EventsSkippedOldSequence</td>
<td>1</td>
<td>The number of incoming events skipped because their sequence indicates they were previously processed.</td>
</tr>
<tr>
<td>Extract</td>
<td>Extract Capture Consumer</td>
<td>InputSchemaRecordsRead</td>
<td>10</td>
<td>The current number of input records of type Schema consumed by this process.</td>
</tr>
<tr>
<td>Extract</td>
<td>Extract Capture Consumer</td>
<td>InputDMLRecordsRead</td>
<td>4728</td>
<td>The current number of input records of type DML (inserts, updates, deletes and truncates) consumed by this process.</td>
</tr>
</tbody>
</table>
CHAPTER 9

Administering PowerExchange CDC Publisher

This chapter includes the following topics:

- Administration Overview, 55
- Reporting the Avro Format Definitions for Source Tables, 56
- Handling Changes to Source Tables and Extraction Maps, 56
- Adding a Source Table to a Change Data Stream, 58
- Encrypting a Password, 59
- Shutting Down the CDC Publisher, 59
- Restarting a PowerExchange CDC Publisher Change Data Stream, 60

Administration Overview

To perform administrative tasks in PowerExchange CDC Publisher, you primarily use the PwxCDCAdmin utility. Check that your file system security allows you to have access to the PwxCDCAdmin utility script.

This section covers the following PwxCDCAdmin administrative tasks:

- Reporting the current Avro format definitions in a legible format, which can be used for validation and by consumer applications
- Handling changes to the source table structure, such as added or dropped columns
- Adding a source table to the change stream of a PowerExchange CDC Publisher process
- Encrypting a password for use in the pwxNodePwd or pwxXmapPwd property in the cdcPublisherPowerExchange.cfg configuration file
- Shutting down a PowerExchange CDC Publisher process

This section also describes how to restart the PwxCDCPublisher utility after it shuts down or ends abnormally using an override restart position in the PowerExchange Logger log records.
Reporting the Avro Format Definitions for Source Tables

Use the PwxCDCAdmin utility with the REPORT=FORMAT parameter to generate the Avro format definitions, or schemas, for all source tables or a specific source table in the \instance\reports subdirectory.

The generated format definitions are available in a legible format for viewing and can be used by consumer applications to read the messages that the PowerExchange CDC Publisher sent to the target messaging system. The generated schemas reflect the flat, nested, or generic format that you set in the Formatter.avroSchemaFormat parameter setting in the cdcPublisherAvro.cfg configuration file.

Before you issue the PwxCDCAdmin command, ensure that the PowerExchange CDC Publisher process is running in a separate command line window.

1. At the command prompt, enter the PwxCDCAdmin command with the REPORT=FORMAT and TABLE parameters.
   You can issue the command from the pwxccpub_root_installation\bin subdirectory or from another location if you specified the PWXPUB_HOME environment variable.
   In the TABLE parameter, indicate whether to generate the schemas for all tables or a specific table. Also include the optional INSTANCE parameter if you are running the PwxCDCPublisher process using an instance other than the default "instanceA" at /pwxccpub_root_installation/instanceA.
   Use the following syntax:
   
   PwxCDCadmin REPORT=FORMAT TABLE=(ALL|mapname_tablename) [INSTANCE=override_instance_directory]

   In this syntax, mapname_tablename is the map name and table name portion of the extraction map name, where the extraction map name has the format xxinstance.mapname_tablename. For more information about the command syntax, see "PwxCDCAdmin Utility - Command and Parameters" on page 66.

2. To verify that the Avro format definitions were generated for the source tables or table, check the reports subdirectory under the instance that you are using.

Handling Changes to Source Tables and Extraction Maps

PowerExchange CDC Publisher does not update the Avro schemas if the source table structure and extraction map definition change. The extractions map definition must change if the source columns or fields for which change data is extracted change, for example, columns are added or removed.

Use the following procedure to add or remove source columns or fields and resume PowerExchange CDC Publisher processing.

1. On the source system, stop change activity (inserts, updates, and deletes) on the source table.
2. Verify that any change data that was captured based on the old source table definition has been extracted from the PowerExchange Logger for Linux, UNIX, and Windows log files.
3. Shut down the PwxCDCPublisher process by using the PwxCDCAdmin SHUTDOWN command. For more information, see "Shutting Down the CDC Publisher" on page 59.
You can issue the command from the pwxcdcpub_root_installation\bin subdirectory or from another location if you specified the PWXPUB_HOME environment variable. For more information, see "Shutting Down the CDC Publisher" on page 59.

   For more information, see the PowerExchange CDC Guide for Linux, UNIX, and Windows.

5. In the PowerExchange Navigator, open the original capture registration and set its status to History.
   For more information, see the PowerExchange Navigator User Guide.

6. Alter the source table or object to add or remove columns.

7. In the PowerExchange Navigator, re-create the capture registration and associated extraction map for the source table. Activate the capture registration.

8. If you set any PowerExchange CDC Publisher properties that refer to the mapname_tablename portion of the extraction map name, update the properties to refer to the mapname_tablename portion of the regenerated extraction map name.
   **Note:** The extraction map name has the format xnninstance.mapname_tablename.
   For example, perform this step if you defined the Extract.captureConsumerExcludeNameList and Extract.captureConsumerIncludeNameList filter properties.

9. On the PowerExchange CDC Publisher system, restart the PwxCDCPublisher process using the RESTART=FROM_END parameter. For more information, see "Restarting a PowerExchange CDC Publisher Change Data Stream" on page 60.

   If you restart the CDC Publisher using the default restart behavior, the FROM_BEGINNING option, or the FROM_CONFIG option, the PowerExchange CDC Publisher issues an error when it tries to process change records based on the previous schema.

10. To clear the old Avro schemas from cache, enter the PwxCDCAdmin command with the CLEAR=FORMAT and TABLE parameters at the command prompt.
    The TABLE parameter is required. Include the optional INSTANCE parameter if you are using an instance other than the default "instanceA" at /pwxcdcpub_root_installation/instanceA.

    Use the following syntax:
    ```
    PwxCDCAdmin CLEAR=FORMAT TABLE=[ALL|mapname_tablename] [INSTANCE=override_instance_directory]
    ```

    In this syntax, mapname_tablename is the map name and table name portion of the new extraction map name, where the extraction map name has the format xnninstance.mapname_tablename. For more information about the command syntax, see "PwxCDCAdmin Utility - Command and Parameters" on page 66.

    The command produces output such as the following example output:
    ```
    INSTANCE = instanceA. Connecting to host:port USHOST1:26987
    --------------------------------------------
    COMMAND  RESULT  RESULT CODE  DESCRIPTION
    CLEAR_FORMAT  10  The number of Avro Schema definitions cleared
    ```

11. To retrieve extraction map information for all source tables from PowerExchange again, including the regenerated extraction map for the updated table, enter the PwxCDCAdmin command with the RESET=FORMAT parameter at the command prompt. Include the optional INSTANCE parameter if you are using an instance other than the default "instanceA" at /pwxcdcpub_root_installation/instanceA.

    Use the following syntax:
    ```
    PwxCDCAdmin RESET=FORMAT [INSTANCE=override_instance_directory]
    ```

    Alternatively, you can wait until the CDC Publisher first receives change data for the updated table. At that point, the CDC Publisher automatically caches the extraction map and generates the Avro schema.
12. If you specified the RESET=FORMAT command and want to force the generation of the updated Avro schema so that you can verify that it properly reflects the changed table structure, enter the PwxCDCAdmin command with the REPORT=FORMAT and TABLE parameters.

The TABLE parameter is required. Include the optional INSTANCE parameter if you are using an instance other than the default “instanceA” at /pwxcdcpub_root_installation/instanceA.

Use the following syntax:

\[\text{PwxCDCAdmin [REPORT=FORMAT TABLE=\{ALL\|mapname_tablename\}] [INSTANCE=override_instance_directory]}\]

In this syntax, mapname_tablename is the map name and table name portion of the new extraction map name.

This command generates new Avro schema in the \instance\reports subdirectory. The schema is generated in a legible format. The generated schema can be used for verification purposes and by consumer applications.

13. Warm start the PowerExchange Logger.

---

**Adding a Source Table to a Change Data Stream**

To add a new source table or object to a PowerExchange CDC Publisher change data stream, you must create a PowerExchange extraction map definition for the new table and restart the PwxCDCPublisher process. Optionally, you can generate an Avro format definition (schema) for the table so that you can verify it.

**Note:** If you do not generate the Avro schema, the CDC Publisher will automatically generate it when change data is received for the new source table.

1. Shut down the PwxCDCPublisher process by issuing the PwxCDCAdmin SHUTDOWN command.

   You can issue the command from the pwxcdcpub_root_installation\bin subdirectory or from another location if you specified the PWXPUB_HOME environment variable. For more information, see "Shutting Down the CDC Publisher" on page 59.

2. Shut down the PowerExchange Logger for Linux, UNIX, and Windows.

   For more information, see the PowerExchange CDC Guide for Linux, UNIX, and Windows.

3. In the PowerExchange Navigator, create a capture registration and associated extraction map for the new source table. Activate the capture registration.

   For more information, see the PowerExchange Navigator User Guide. In the capture registration, set the Status to Active and the Condense option to Part.

4. Restart the PwxCDCPublisher process. For more information, see "Restarting a PowerExchange CDC Publisher Change Data Stream" on page 60.

5. To retrieve extraction map information for all source tables from PowerExchange, including the new table, enter the PwxCDCAdmin command with the RESET=FORMAT parameter at the command prompt. Include the optional INSTANCE parameter if you are using an instance other than the default “instanceA” at /pwxcdcpub_root_installation/instanceA.

   Use the following syntax:

   \[\text{PwxCDCAdmin RESET=FORMAT [INSTANCE=override_instance_directory]}\]

   Alternatively, you can wait until the CDC Publisher first receives change data for the new table. At that point, the CDC Publisher automatically caches the extraction map and generates the Avro schema.
6. If you specified the PwxCDCAdmin RESET=FORMAT command and want to force the generation of the Avro schema for the new table so that you can verify that it properly reflects the table structure, enter the PwxCDCAdmin command with the REPORT=FORMAT and TABLE parameters at the command prompt. The TABLE parameter is required. Include the optional INSTANCE parameter if you are using an instance other than the default "instanceA" at /pxcwdcpub_root installation/instanceA.

Use the following syntax:

```
PwxCDCAdmin REPORT=FORMAT TABLE=mapname_tablename[INSTANCE=override_instance_directory]
```

In this syntax, `mapname_tablename` represents the table name portion of the new extraction map name, where the extraction map name has the format `xnninstance.mapname_tablename`.

This command generates the updated Avro schema in the `\instance\reports` subdirectory. The schema is generated in a legible format. The generated schema can be used for verification purposes and by consumer applications.

7. Warm start the PowerExchange Logger.

Encrypting a Password

You can encrypt a password to be entered in the Extract.pwxmapPwd or Extract.pwxNodePwd property in the cdcPublisherPowerExchange.cfg configuration text file to protect the password from unauthorized viewing. Use the PwxCDCAdmin utility with the ENCRYPT parameter.

**Note:** Do not use this method to encrypt a password that will be entered in the Extract.pwxNodeEPwd property. This property value must be encrypted with the PowerExchange Navigator Encrypt Password facility.

- At the command prompt, enter the PwxCDCAdmin command with the ENCRYPT parameter.

Use the following syntax:

```
PwxCDCAdmin ENCRYPT=password_value
```

You can issue the command from the `pxcwdcpub_root installation\bin` subdirectory or from another location if you specified the PWXPUB_HOME environment variable.

The PwxCDCAdmin utility generates and displays an encrypted password in the format `ENC:encrypted_value`, where "ENC:" is the prefix that the CDC Publisher uses to identify an encrypted password value.

Shutting Down the CDC Publisher

Use the PwxCDCAdmin utility with the SHUTDOWN parameter to stop the PowerExchange CDC Publisher process.

If messages are being written to topics when the shutdown request is made, the connection to the target messaging system remains open so that the write operations can complete before the CDC Publisher shuts down. If CDC Publisher internal processes are processing change data, the processes ignore the in-progress changes and shut down cleanly. The processing of these in-progress changes will occur after the CDC Publisher process is restarted, provided that you use the default restart position.
Before you issue the PwxCDCAdmin command, ensure that the PowerExchange CDC Publisher process is running in a separate command line window.

▶ At the command prompt, enter the PwxCDCAdmin command with the SHUTDOWN parameter. Include the optional INSTANCE parameter if you are shutting down a PwxCDCPublisher process that is using an instance other than the default "instanceA" at /pwxcdcpub_root installation/instanceA.

Use the following syntax:

```
PwxCDCAdmin SHUTDOWN [INSTANCE=override_instance_directory]
```

You can issue the command from the pwxcdcpub_root installation\bin subdirectory or from another location if you specified the PWXPUB_HOME environment variable. For more information about the command syntax, see "PwxCDCAdmin Utility - Command and Parameters" on page 66.

The following output from shutdown processing is displayed:

```
INSTANCE = instanceA. Connecting to host:port USHOST1:26987
---------------------------------------------------------------
Shutdown request has been executed...
Waiting for server to shutdown. Wait duration 0:00:11 as of 2017/09/22 11:29:19
Shutdown completed. Server connection is no longer available.
```

## Restarting a PowerExchange CDC Publisher Change Data Stream

If a PowerExchange CDC Publisher process is shut down or ends for any reason, you can use the PwxCDCPublisher utility to restart it from the default restart position in the PowerExchange Logger for Linux, UNIX, and Windows log files or from an override restart position that you set by using the PwxCDCPublisher RESTART or COLDSTART parameter.

By default, the first time you start the PwxCDCPublisher for a particular instance, it starts from the earliest position in the PowerExchange Logger log files. Subsequent PwxCDCPublisher runs start from the last position of the prior run unless you specify an override position. If you plant to override this default position by using the RESTART FROM_BEGINNING or FROM_CONFIG option or the COLDSTART=Y setting, first delete the existing checkpoint file.

▶ At the command prompt, enter the PwxCDCPublisher command without the RESTART or COLDSTART parameter to restart the Publisher process from the default restart position. To override the default restart position, include the optional RESTART or COLDSTART parameter. Also include the optional INSTANCE parameter if you are restarting a PwxCDCPublisher process that is using an instance other than the default "instanceA" at /pwxcdcpub_root installation/instanceA.

You can issue the command from the pwxcdcpub_root installation\bin subdirectory or from another location if you specified the PWXPUB_HOME environment variable.

Use the following syntax:

```
PWXCDCPUBLISHER [RESTART={FROM_BEGINNING|FROM_END|FROM_CONFIG}] [COLDSTART={Y|N}] [INSTANCE=override_instance_directory]
```
In this syntax:

- **RESTART.** Overrides the default restart position. Select one of the following options to specify the location in the change stream from which to restart Publisher processing:
  - **FROM_BEGINNING.** Extract change data from the earliest record that is available in the PowerExchange Logger log files, which is the same as the default start position for the first run of the PowerExchange CDC Publisher.
  - **FROM_END.** Extract change beginning from the last processed record in the PowerExchange Log log files.
  - **FROM_CONFIG.** Extract change data from the specific change stream position that you identify in the Extract.restart1 and Extract.restart2 properties in the cdcPublisherPowerExchange.cfg configuration file. This option is equivalent to setting COLDSTART=Y.

  **Note:** The Extract.restart1 and Extract.restart2 properties specify the PowerExchange sequence token and restart token values, respectively. For more information about these PowerExchange token values, see the PowerExchange CDC documentation or contact Informatica Global Customer Support.

- **COLDSTART.** Set this parameter to Y to cold start the PwxCDCPublisher process from the specific position in the PowerExchange Logger log files that is identified by the Extract.restart1 and Extract.restart2 properties. This setting is equivalent to RESTART=FROM_CONFIG. Do not specify the COLDSTART=Y parameter with the RESTART parameter.

For more information about the command syntax, see "PwxCDCAdmin Utility - Command and Parameters" on page 66.

The following example output is from a PWXCDCPublisher RESTART=FROM_END command:

```
INSTANCE = instanceA. CDC Publisher Started. Listening on host:port USHOST1:26987
---------------------------------------------------------------------------
Running...
Pwx-073304 Connection: 10.17.3.191:49363, Type: CAXRT, Task ID: 0, Operation timeout: (-) secs, Heartbeat interval: 30 secs, Session: (-)
Pwx-033304 Connection: 10.17.3.191:49363, Type: CAXRT, Task ID: 0, Operation timeout: (-) secs, Heartbeat interval: 30 secs, Session: (-)
Pwx-073304 Connection: 10.17.3.191:49363, Type: CAXRT, Task ID: 0, Operation timeout: (-) secs, Heartbeat interval: 30 secs, Session: (-)
Pwx-073304 Connection: 10.17.3.191:49363, Type: CAXRT, Task ID: 0, Operation timeout: (-) secs, Heartbeat interval: 30 secs, Session: (-)
Pwx-073304 Connection: 10.17.3.191:49363, Type: CAXRT, Task ID: 0, Operation timeout: (-) secs, Heartbeat interval: 30 secs, Session: (-)
Pwx-073304 Connection: 10.17.3.191:49363, Type: CAXRT, Task ID: 0, Operation timeout: (-) secs, Heartbeat interval: 30 secs, Session: (-)
```
Appendix A

Command Reference for the Command-Line Utilities

This appendix includes the following topics:

- Command Reference Overview, 62
- PwxCDCPublisher Utility - Command and Parameters, 63
- PwxCDCInfo Utility - Command and Parameters, 64
- PwxCDCAdmin Utility - Command and Parameters, 66

Command Reference Overview

This command reference describes the proper syntax, usage, and limitations of commands for controlling the PowerExchange CDC Publisher command-line utilities.

Commands for the following PowerExchange CDC Publisher utilities are covered:

- PwxCDCPublisher utility - The main utility for streaming change data from PowerExchange to the target messaging system. The utility extracts change data from PowerExchange, filters the data by source object if filtering criteria is defined, formats the data into messages, connects to the target, and publishes the messages to the target. The utility initiates a data stream path that is composed of all of these processes.
- PwxCDCAdmin utility - The utility for issuing administrative commands to an active PwxCDCPublisher process.
- PwxCDCInfo utility - The utility for reporting the status of an active PwxCDCPublisher process and each of its main subprocesses, statistics and state information for each target topic, and internal attributes for diagnostic use.

The following table describes the conventions that the command reference uses to indicate the proper command syntax:

<table>
<thead>
<tr>
<th>Convention</th>
<th>Description</th>
<th>Example</th>
</tr>
</thead>
<tbody>
<tr>
<td>Monospaced font</td>
<td>Indicates the command syntax or a command example.</td>
<td>PWXCDCPUBLISHER HELP</td>
</tr>
<tr>
<td><em>Italics</em></td>
<td>Indicates a variable in the syntax for which you enter a specific value.</td>
<td>[ENCRIPT=password_value]</td>
</tr>
</tbody>
</table>
To run a command, navigate to the `bin` subdirectory that contains the utility script. Then enter the script name with the appropriate parameters at the command line.

You can run the PwxCDCAdmin and PwxCDCInfo utilities on a system that is remote from host where the PowerExchange CDC Publisher process is running. In this scenario, you must install the PowerExchange CDC Publisher on the remote system as well as on the CDC Publisher system. Also, in the configuration file on the remote system, you must set the Common.pwxCDCPublisherHost property to the public IP address of the CDC Publisher system and define the PWXPUB_HOME environment variable.

### PwxCDCPublisher Utility - Command and Parameters

Use the PwxCDCPublisher utility to create and run a change data stream from PowerExchange to a supported target.

Also, the PwxCDCPublisher utility has optional parameters that allow you to specify an instance subdirectory other than instanceA and to indicate the change stream position from which the PwxCDCPublisher will restart.

#### Syntax

The PwxCDCPublisher command has the following syntax:

```
PWXCDCPUBLISHER
  [INSTANCE=override_instance_directory]
  [RESTART={FROM_BEGINNING|FROM_END|FROM_CONFIG}]
  [COLDSTART={Y|N}]
  [HELP=?]
```

All parameters are optional. Use a single space as a separator between parameters.

#### Parameter Descriptions

**INSTANCE=override_instance_subdirectory**

Overrides the default name or location of the instance subdirectory, which is `pwxcdcpub_root installation\instanceA`. You can enter the fully qualified path of the subdirectory or a location that is relative to the root installation directory that is specified in the PWXPUB_HOME environment variable. For example, the following values specify the same directory location if PWXPUB_HOME=C:\Informatica\PwxCdcPublisherInstall.1.0.0.windows.64bit:

- `instanceB`
- `C:\Informatica\PwxCdcPublisherInstall.1.0.0.windows.64bit\instanceB`
RESTART=[{FROM_BEGINNING|FROM_END|FROM_CONFIG}]

Overrides the default change stream position from which the PwxCDCPublisher process restarts after it
is stopped or ends abnormally. To indicate the change stream position, enter one of the following
options:

- FROM_BEGINNING. Extract change data from the earliest record that is available in the
  PowerExchange Logger log files.
- FROM_END. Extract change beginning from the last processed record in the PowerExchange Log log
  files.
- FROM_CONFIG. Extract change data from the specific change stream position that you identify in the
  Extract.restart1 and Extract.restart2 properties in the Extractor configuration file. This option is
  equivalent to setting COLDSTART=Y.

Note: The Extract.restart1 and Extract.restart2 properties specify the PowerExchange sequence token
and restart token values, respectively. For more information about these PowerExchange token
values, see the PowerExchange CDC documentation or contact Informatica Global Customer Support.

By default, the first time you start the PwxCDCPublisher, it starts from the earliest position in the
PowerExchange Logger log files, as if FROM_BEGINNING was specified. Subsequent PwxCDCPublisher
runs start from the last position of the prior run. If you want it to start from a different position, use the
RESTART parameter. Do not also specify the COLDSTART parameter.

COLDSTART={Y|N}

Indicates whether to cold start the PwxCDCPublisher process. If you specify COLDSTART=Y, the process
begins extraction processing from the change stream position that you specify in the Extract.restart1
and Extract.restart2 properties in the cdcPublisherPowerExchange.cfg configuration file. The
COLDSTART=Y setting is equivalent to the RESTART=FROM_CONFIG setting. COLDSTART=N indicates a
cold start will not occur.

Note: Issue either the COLDSTART or RESTART parameter. Do not specify both parameters.

Default value is N.

HELP or ?

Displays help information about the PwxCDCPublisher parameters.

PwxCDCInfo Utility - Command and Parameters

Use the PwxCDCInfo utility to display summary or detailed status information and internal processing
statistics for a PwxCDCPublisher process.

Tip: The STATS INTERNAL and STATUS DETAIL output is displayed in a wide format. For easier reading,
direct the output to a file.

Syntax

The PwxCDCInfo command has the following syntax:

PWXCDCINFO
   [INSTANCE=override_instance_directory]  
   [STATS={TOPIC|INTERNAL [VERBOSE]}][RESET]
   [STATUS [DETAIL [VERBOSE]]]

All parameters are optional. Use a single space as a separator between parameters.
Parameter Descriptions

INSTANCES=override_instance_subdirectory

Overrides the default name or location of the instance subdirectory, which is \pwxcdcpub_root \installation\instanceA. You can enter the fully qualified path of the subdirectory or a location that is relative to the root installation directory that is specified in the PWXPU.Home environment variable. For example, the following values specify the same directory location if PWXPU.Home=C:\Informatica\PwxCDCPublisherInstall.1.0.0.windows.64bit:

instanceB
C:\Informatica\PwxCDCPublisherInstall.1.0.0.windows.64bit\instanceB

STATS=(TOPIC|INTERNAL [VERBOSE]) [RESET]

Reports information about the topics that the CDC Publisher has updated or internal statistics and state information for PowerExchange CDC Publisher child processes (Extract, Formatter, and Connector). The statistics are cumulative since the Publisher process started. You can use the optional RESET parameter to refresh the topic information or reset the statistics counters to their original values.

Do not specify both the TOPIC and INTERNAL parameters.

You can enter the STATS parameter in any of following ways:

- **STATS=TOPIC**. Reports the following information for each topic that the CDC Publisher updated:
  - Updated topic name and partition ID
  - Number of messages sent to the topic
  - Partition offset returned
  - Source table name
  - Last change operation type and sequence ID

- **STATS=TOPIC RESET**. Clears the statistics for the updated topics.

- **STATS=INTERNAL**. Reports detailed statistics and state information, by internal attribute, for each type of PowerExchange CDC Publisher child process. This information is primarily for diagnostic use by Informatica Global Customer Support and other technical staff. The following example shows the report format:

<table>
<thead>
<tr>
<th>PARENT</th>
<th>CHILD</th>
<th>ATTRIBUTE</th>
<th>VALUE</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>Extract</td>
<td>Extract Capture Consumer</td>
<td>InputRecordsRead</td>
<td></td>
<td></td>
</tr>
<tr>
<td>2018</td>
<td>Extract Capture Consumer</td>
<td>InputBytesRead</td>
<td>12560180</td>
<td></td>
</tr>
</tbody>
</table>

- **STATS=INTERNAL VERBOSE**. Reports detailed statistics and state information, by internal attribute, for each of the major PowerExchange CDC Publisher internal processes with a description of the attribute. The following example shows the verbose report format:

<table>
<thead>
<tr>
<th>PARENT</th>
<th>CHILD</th>
<th>ATTRIBUTE</th>
<th>VALUE</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>Extract</td>
<td>Extract Capture Consumer</td>
<td>InputRecordsRead</td>
<td>2018</td>
<td>The current number of input records consumed by this process.</td>
</tr>
<tr>
<td></td>
<td>Extract Capture Consumer</td>
<td>InputBytesRead</td>
<td>12560180</td>
<td>The current number of bytes of input consumed by this process.</td>
</tr>
</tbody>
</table>

Note: Do not specify INTERNAL VERBOSE with the RESET parameter.

- **STATS=INTERNAL RESET**. Resets the internal attribute values to their initial values at CDC Publisher startup.

**STATUS [DETAIL [VERBOSE]]**

Indicates the status of the PwxCDCPublisher processes.
You can enter the STATUS parameter in any of following ways:

- **STATUS.** Reports the overall status of all PowerExchange CDC Publisher processes by using a color indicator and brief description. For example:

  ```
  PROCESS STATE      GREEN     All processes are running
  ```

  Other process states are: YELLOW for a warning situation, or RED for a CDC Publisher subprocess that stopped or could not be found. If the main CDC Publisher process terminates or is unavailable for connection, error messages are reported instead of a process state.

- **STATUS DETAIL.** Reports the status color indicator for each of the major PowerExchange CDC Publisher internal processes, including the Extract, Filter, Formatter, and Connector processes. The following example shows the status of the Extract internal process:

  ```
  PARENT    CHILD     ATTRIBUTE     VALUE     DESCRIPTION
  Extract_Connector Extract TRAFFIC LIGHT GREEN
  ```

  **Note:** VERBOSE can be specified only with the DETAIL parameter.

---

**PwxCDCAdmin Utility - Command and Parameters**

Use the PwxCDCAdmin utility to perform administrative functions for the PowerExchange CDC Publisher process.

With the PwxCDCAdmin command and its parameters, you can perform the following tasks:

- Report the Avro schemas, or format definitions, for all source tables or a selected source table for viewing or consumer application use.
- Clear the generated Avro schemas for source tables from cache. The CDC Publisher uses these schemas to format data in the messages to be sent to the target.
- Re-read extraction map definitions for all data sources from PowerExchange and cache any new and updated information.
- Shut down the PwxCDCPublisher process after it completes current work in progress.

**Tip:** You can use file-system security to limit rights on the PwxCDCAdmin utility script.

**Syntax**

```plaintext
PWXCDCADMIN
[INSTANCE=override_instance_directory]
[REPORT=FORMAT TABLE={ALL|mapname_tablename}]
[CLEAR=FORMAT TABLE={ALL|mapname_tablename}]
[RESET=FORMAT]
[ENCRYPT=password_value]
[SHUTDOWN]
```

All parameters are optional. Use one or more spaces as the separator between parameters.
Parameter Descriptions

**INSTANCE=override_instance_subdirectory**

Overrides the default name or location of the instance subdirectory, which is `pwxcddcpub_root\installation\instance\`. You can enter the fully qualified path of the subdirectory or a location that is relative to the root installation directory that is specified in the PWXPUB_HOME environment variable. For example, the following values specify the same directory location if PWXPUB_HOME=C:\Informatica\PwxCdcPublisherInstall.1.0.0.windows.64bit:

```
\instanceB
C:\Informatica\PwxCdcPublisherInstall.1.0.0.windows.64bit\instanceB
```

**REPORT=FORMAT TABLE={ALL|mapname_tablename}**

Reports the existing Avro schemas, or format definitions, for either all source tables or a selected source table in the `/pwxcddcpub_root\installation\instance\reports` directory. These Avro schemas are produced in a legible format. The schemas can be used for validation purposes or by consumer applications. If no Avro schemas exist for one or more source tables, the utility generates the schemas based on the Formatter properties in the cdcPublisherAvro.cfg configuration file and cached extraction map definitions. If the extraction maps are not in cache, you can use the RESET parameter to re-read them from PowerExchange.

The TABLE parameter is required and has no default value. Set the TABLE parameter to either ALL or a specific `mapname_tablename` value from an extraction map name. If this value includes special characters, the CDC Publisher replaces the special characters with the replacement character that is specified in the Formatter.avroNameReplacementValue parameter. Avro does not support special characters.

**Tip:** To determine the `mapname_tablename` value, look up the extraction map name in the PowerExchange Navigator. The extraction map name has the format `xnninstance.mapname_tablename`. Use the `mapname_tablename` portion in the TABLE parameter. If the extraction map name does not include the `tablename`, change records have not yet been extracted based on the extraction map. For more information, see the PowerExchange Navigator User Guide.

**CLEAR=FORMAT TABLE={ALL|mapname_tablename}**

Clears the Avro schemas, or format definitions, for all source tables or a selected source table from cache. Use this parameter if the Avro format definition for one or more source tables has changed and is no longer valid for CDC Publisher processing. When the CDC Publisher next receives change data for a changed source table, it regenerates the Avro schema. Alternatively, you can force the Avro schema to be regenerated by using the REPORT=FORMAT parameter.

The TABLE parameter is required and has no default value. Set the TABLE parameter to either ALL or a specific `mapname_tablename` value from an extraction map name.

**RESET=FORMAT**

Re-reads extraction map definitions for all data sources from PowerExchange and caches them. PowerExchange CDC Publisher uses the extraction map information to generate new Avro schemas when either change data is next received for the source tables associated with the extraction maps or when you issue the PwxCDCAdmin command with the REPORT=FORMAT parameter.

If you add a source table, you can use the RESET=FORMAT parameter to get the extraction map definition for the new table right away so that you can generate the Avro schema for verification purposes, before change data is received from the table. Alternatively, you can wait until change data from the new table is first received. At that point, the CDC Publisher caches the extraction map and generates the Avro schema.
**ENCRYPT=password_value**

Encrypts a password value to be used in the Extract.pwxXmapPwd or Extract.pwxNodePwd property in the cdcPublisherPowerExchange.cfg configuration file to protect the password from unauthorized viewing. The PwxCDCAdmin utility generates and displays encrypted passwords in the following format:

```
ENC:encrypted_value
```

Where "ENC:" is the prefix that the CDC Publisher uses to identify an encrypted password value.

Do not use this type of encryption for the Extract.pwxNodeEPwd property value. For this property, use the PowerExchange Navigator Encrypt Password facility to encrypt the password. For more information, see the *PowerExchange Navigator User Guide*.

**SHUTDOWN**

Shuts down the PowerExchange CDC Publisher process cleanly. If messages are being written to topics when the shutdown request is made, the connection to Kafka target remains open until all of the sent messages are acknowledged by Kafka as received. Then the CDC Publisher shuts down.
This appendix includes the following topics:

- **Types of Avro Schema Formats, 69**
- **Avro Fields That Define a Source Column, 70**
- **Avro Flat Schema Format, 70**
- **Avro Nested Schema Format, 71**
- **Avro Generic Schema Format, 71**
- **Avro Wrapper Schema Format, 71**

### Types of Avro Schema Formats

PowerExchange CDC Publisher generates an Avro schema for each source object. The Avro schema defines the structure of the messages to be sent to Apache Kafka topics. Consumer applications must have a copy of the Avro schema to consume the Avro-formatted message content.

In the PowerExchange CDC Publisher cdcPublisherAvro.cfg configuration file, you specify the Avro schema format type in the Formatter.avroSchemaFormat parameter. Options are:

- avroFlatSchemaFormatV1 - The "flat" schema
- avroNestedSchemaFormatV1 - The "nested" schema
- avroGenericSchemaFormatV1 - The "generic" schema

Also, you can "wrap" a flat, nested, or generic schema in a "wrapper" schema by setting the Formatter.avroWrapperSchemaFormatV1 property to avroWrapperSchemaFormatV1 in the cdcPublisherAvro.cfg configuration file. The wrapper schema defines three fields for the following metadata: the change sequence number, the source `mapname_tablename` value from the extraction map name, and the change operation type. The wrapper schema also includes a field for the "wrapped" child schema, which is expressed as a large string.

A generic or wrapper schema can be used for multiple source objects. Consider using a generic or wrapper schema when you want to send messages with change data from multiple source objects to a single Kafka topic. The Kafka topic is identified in the Connector.kafkaTopic property.

A flat or nested schema pertains to a specific source table. It defines the columns in the source table. Consider using a flat or nested schema when you want to send change data from a specific source object to the Kafka topic that is generated for that source object. In this case, set the Connector.kafkaTopic property to USE_TABLE_NAME.

This appendix shows the general structure of the flat, nested, generic, and wrapper formats. It also describes the four Avro fields that are used to describe each source column or field in an Avro flat or nested schema.
Avro Fields That Define a Source Column

In an Avro flat or nested schema, the following four fields define each source column:

- **Column definition:**
  ```json
  {"name": "COL1", "type": ["null", "string"], "doc": "...", "default": null}
  ```

- **Column value is-present indicator:**
  ```json
  {"name": "COL1_Present", "type": "boolean", "doc": "Contains true when the value of field COL1 is present in the change data stream", "default": false}
  ```

- **Column before image:**
  ```json
  {"name": "COL1_BeforeImage", "type": ["null", "string"], "doc": "Contains the before image of field COL1", "default": null}
  ```

- **Column before image is-present indicator:**
  ```json
  {"name": "COL1_BeforeImage_Present", "type": "boolean", "doc": "Contains true when the value of field COL1_OLD is present.", "default": false}
  ```

For example, two columns named 'COL1' and 'COL2' have the following Avro definitions:

```json
{"name": "COL1", "type": ["null", "string"], "doc": "...", "default": null}
{"name": "COL1_Present", "type": "boolean", "doc": "..."}
{"name": "COL1_BeforeImage", "type": ["null", "string"], "doc": "..."}
{"name": "COL1_BeforeImage_Present", "type": "boolean", "doc": "..."}

{"name": "COL2", "type": ["null", "string"], "doc": "...", "default": null}
{"name": "COL2_Present", "type": "boolean", "doc": "..."}
{"name": "COL2_BeforeImage", "type": ["null", "string"], "doc": "..."}
{"name": "COL2_BeforeImage_Present", "type": "boolean", "doc": "..."}
```

Avro Flat Schema Format

An Avro schema that uses the flat format contains all of the Avro column-definition fields in one Avro record. The schema is specific to the source object that contains the defined columns.

The following flat schema defines two columns in one record:

```json
{"type": "record", "fields": [ 
  {"name": "COL1", "type": ["null", "string"], "doc": "...", "default": null}
  {"name": "COL1_Present", "type": "boolean", "doc": "..."}
  {"name": "COL1_BeforeImage", "type": ["null", "string"], "doc": "..."}
  {"name": "COL1_BeforeImage_Present", "type": "boolean", "doc": "..."}
  {"name": "COL2", "type": ["null", "string"], "doc": "...", "default": null}
  {"name": "COL2_Present", "type": "boolean", "doc": "..."}
  {"name": "COL2_BeforeImage", "type": ["null", "string"], "doc": "..."}
  {"name": "COL2_BeforeImage_Present", "type": "boolean", "doc": "..."}
]}
Avro Nested Schema Format

An Avro schema that uses the nested format has a main Avro record that contains a separate nested record for each type of Avro column-definition field. The schema is specific to the source object that contains the defined columns.

The following nested schema defines two columns in four records:

```json
{  "name": "columns",  "type": "record",  "fields": [    {      "name": "COL1",      "type": [        "null",        "string"      ],      "doc": "...",      "default": null    },    {      "name": "COL2",      "type": [        "null",        "string"      ],      "doc": "...",      "default": null    }  ]}
```

Avro Nested Schema Format

An Avro schema that uses the nested format has a main Avro record that contains a separate nested record for each type of Avro column-definition field. The schema is specific to the source object that contains the defined columns.

The following nested schema defines two columns in four records:

```json
{  "name": "columns",  "type": [    {      "name": "COL1",      "type": [        "null",        "string"      ],      "doc": "...",      "default": null    },    {      "name": "COL2",      "type": [        "null",        "string"      ],      "doc": "...",      "default": null    }  ]}
```

Avro Generic Schema Format

An Avro schema that uses the generic format can be used for any source table. The schema represents the source columns as an array of fields for data and metadata. Most of the columns have the type 'string' to be generic across multiple source table types. The column name and type is included in every Avro message along with the data.

A generic schema has the following format:

```json
{  "name": "columns",  "type": [    {      "name": "value",      "type": [        "null",        "string"      ],      "doc": "...",      "default": null    },    {      "name": "isPresent",      "type": [        "null",        "boolean"      ],      "doc": "...",      "default": null    },    {      "name": "beforeImageValues",      "type": [        "null",        "string"      ],      "doc": "...",      "default": null    },    {      "name": "isPresentIndicatorsForBeforeImages",      "type": [        "null",        "string"      ],      "doc": "...",      "default": null    }  ]}
```

Avro Wrapper Schema Format

An Avro "wrapper" schema wraps a flat, nested, or generic schema. The wrapper, or parent, schema defines three fields for metadata and one field for the wrapped child schema, which is expressed as a large string. The wrapped child schema has the format that is specified in the Formatter.avroSchemaFormat property.

To a consumer application, all messages that use the wrapper schema appear to have the same four-field format. The metadata fields are for the change sequence number, the source mapname tablename value from the extraction map name, and the change operation type.
A single wrapper schema can be used to send messages to Kafka that contain change data from different source objects, using different Avro schema formats. When the consumer application reads the Kafka messages, it parses the underlying data and puts it in the Avro format designated in the wrapped child schema field.

As compared with the Avro generic schema format, a wrapper schema results in smaller messages because it adds only the source `mapname_tablename` in each message instead of both the column name and type. Also, the wrapper schema maintains the column data types rather than using the generic "string" type for all columns with data.

To process the data in Kafka messages based on this schema format, the consumer application must parse the messages to get the source `mapname_tablename` and then find the Avro flat, nested, or generic schema that matches that name value by using their own methods.

A wrapper schema has the following format:

```json
{
  "type" : "record",
  "name" : "InfaAvroWrapperSchema",
  "fields" : [
    {
      "name" : "INFA_SEQUENCE","type" : [ "null", "string" ], "default" : null},
    {
      "name" : "INFA_TABLE_NAME","type" : [ "null", "string" ],"default" : null},
    {
      "name" : "INFA_OP_TYPE","type" : [ "null", "string" ],"default" : null},
    {
      "name" : "ChildSchema","type" : [ "null", "string" ],"default" : null}
  ]
}
```
Message Reference

This appendix includes the following topics:

- **Message Reference Overview, 73**
- CDCPUB_10001 to CDCPUB_16001, 74
- CPPSVC_00100 to CPPSVC_00103, 103
- LOGCFG_10001 to LOGCFG_10006, 104
- PWXPLIC_00110 to PWXPLIC-00999, 105
- PWXRDR-01000 to PWXRDR-01301, 107

Message Reference Overview

This appendix describes the messages that are issued by PowerExchange CDC Publisher.

The message prefixes indicates the message categories. Messages in each of the following categories are described:

<table>
<thead>
<tr>
<th>Message Prefix</th>
<th>Message Category</th>
</tr>
</thead>
<tbody>
<tr>
<td>CDCPUB</td>
<td>CDC Publisher common messages</td>
</tr>
<tr>
<td>CPPSVC</td>
<td>Diagnostic messages for a C++ service that is involved in CDC Publisher extraction of change data from PowerExchange</td>
</tr>
<tr>
<td>LOGCFG</td>
<td>Message-logging configuration messages</td>
</tr>
<tr>
<td>PWXPLIC</td>
<td>Product licensing messages</td>
</tr>
<tr>
<td>PWXRDR</td>
<td>PWX Reader messages</td>
</tr>
</tbody>
</table>

*Note:* The PWX Reader is part of the CDC Publisher Extractor process.
The process [process_name] could not perform the state change or action [action]. Reason: [reason].

Explanation: The specified internal process encountered an error while trying to stop, start, or perform another requested action. This message includes the reason for the error.

User Response: To diagnose and correct the error, analyze the accompanying log messages. If the problem persists, contact Informatica Global Customer Support.

Generating dump output. Reason: [reason].

Explanation: This error message introduces dump information that was produced by the CDC Publisher for the specified reason.

User Response: To diagnose and correct the error, analyze the accompanying log messages. If the problem persists, contact Informatica Global Customer Support.

DUMP: [text].

Explanation: This error message contains one or more lines of detailed debug information, in text format, from a CDC Publisher dump operation.

User Response: To diagnose and correct the error, analyze the accompanying log messages. If the problem persists, contact Informatica Global Customer Support.

HEXDUMP: [hexadecimal_dump_information].

Explanation: This error message contains one or more lines of detailed debug information, in hexadecimal format, from a CDC Publisher dump operation.

User Response: To diagnose and correct the error, analyze the accompanying log messages. If the problem persists, contact Informatica Global Customer Support.

Generated dump complete.

Explanation: This error message identifies the end of the detailed debug information from a CDC Publisher dump operation.

User Response: To diagnose and correct the error, analyze the accompanying log messages. If the problem persists, contact Informatica Global Customer Support.

The process [current_process_name] was unable to start a thread for [failed_process_name]. The maximum number of threads might have been reached. Error: [error_description].

Explanation: The specified process encountered an error while trying to start a new operating system thread. This message includes the thread ID or the name of the subprocess associated with the thread, and a description of the error. This error can be caused by an operating system resource limitation or a failure to remove unused threads.

User Response: To diagnose and correct the error, analyze the accompanying log messages. If the problem persists, contact Informatica Global Customer Support.
**CDCPUB_10007** The process [current_process_name] was unable to start a thread for [failed_process_name] because of an exception. Error: [error_description].

Explanation: The specified process encountered an error while trying to start a new operating system thread. The request for a new thread generated an exception. This problem can occur because of an operating system resource limitation or because the CDC Publisher cannot remove unused threads.

User Response: To diagnose and correct the error, analyze the accompanying log messages. If the problem persists, contact Informatica Global Customer Support.

**CDCPUB_10008** The process [current_process_name] could not process the request named [failed_request_name]. The process current state [state] does not support the request. Error: [error_description].

Explanation: An internal process was unable to stop, start, or perform another requested action because its current state does not support the request type.

User Response: To diagnose and correct the error, analyze the accompanying log messages. If the problem persists, contact Informatica Global Customer Support.

**CDCPUB_10011** The process [process_name] with thread ID [thread_ID] stopped running because of a fatal error.

Explanation: The internal process or its operating system thread stopped unexpectedly because of a fatal error.

User Response: To diagnose and correct the error, analyze the accompanying log messages. If the problem persists, contact Informatica Global Customer Support.


Explanation: The specified internal process or its operating system thread stopped unexpectedly because of an error. This message provides details about the error.

User Response: To diagnose and correct the error, analyze the accompanying log messages. If the problem persists, contact Informatica Global Customer Support.

**CDCPUB_10013** The process [current_process_name] could not find the directory [directory_name] requested in the configuration parameter [parameter_name].

Explanation: The specified configuration parameter points to a directory on the local file system, but the specified process cannot find that directory.

User Response: To correct the error, verify that the configuration parameter points to an existing directory, or create the directory if it does not exist.

**CDCPUB_10014** The process [current_process_name] could not write to the file or directory [file_or_directory_name] requested in the configuration parameter [parameter_name].

Explanation: The specified configuration parameter points to a file or directory that does not exist or is not writable. As a result, the process cannot write to the file or directory.

User Response: To correct the error, verify that the configuration parameter points to an existing file or directory for which write permissions have been granted.
CDCPUB_10015  The process [extractor_process_name] has requested a shutdown of its helper thread [thread_name].
Explanation: This informational message reports that the CDC Publisher Extractor process requested its helper thread to shut down.
User Response: No response is required.

CDCPUB_10016  The process [extractor_process_name] was unable to shut down its helper thread [thread_name].
Explanation: The CDC Publisher Extractor process encountered an error while trying to shut down its internal helper thread.
User Response: To diagnose and correct the error, analyze the accompanying log messages. If the problem persists, contact Informatica Global Customer Support.

CDCPUB_10017  The process [extractor_process_name] is waiting for its helper thread [thread_name] to shut down. Current wait duration: [milliseconds].
Explanation: The CDC Publisher Extractor process requested that its internal helper thread shut down. This informational message reports the number of milliseconds that the process has been waiting for the helper thread to complete shutdown processing.
User Response: No response is required.

CDCPUB_10018  The helper [helper_process_name] with thread ID [thread_ID] is starting.
Explanation: This informational message reports that the specified internal helper process is starting on the specified thread.
User Response: No response is required.

CDCPUB_10019  The helper [helper_process_name] with thread ID [thread_ID] has shut down successfully.
Explanation: This informational message reports that the specified internal helper process has shut down successfully.
User Response: No response is required.

CDCPUB_10020  The helper [helper_process_name] with thread ID [thread_ID] stopped running because of a fatal error.
Explanation: The specified internal helper process stopped because of a fatal error.
User Response: To diagnose and correct the error, analyze the accompanying log messages. If the problem persists, contact Informatica Global Customer Support.

Explanation: The specified internal helper process stopped because of an error. This message provides information about the error.
User Response: To diagnose and correct the error, analyze the accompanying log messages. If the problem persists, contact Informatica Global Customer Support.

CDCPUB_10022  Stack Trace [stack_trace_information].
Explanation: This message reports stack trace information, which is typically for debugging a preceding error.
User Response: To diagnose and correct the error, analyze the accompanying log messages. If the problem persists, contact Informatica Global Customer Support.

**CDCPUB_10023** The process [process_name] result for a test connection request was successful.

Explanation: The specified internal process successfully tested connectivity to an outside element.

User Response: No response is required.

**CDCPUB_10024** The process [process_name] result for a test connection request was not successful.

Explanation: This informational message reports that the specified internal process tested connectivity to an outside element, but the test was unsuccessful.

User Response: No response is required.


Explanation: The internal process encountered an error. Typically, the details of the error are provided in accompanying error messages.

User Response: To diagnose and correct the error, analyze the accompanying log messages. If the problem persists, contact Informatica Global Customer Support.

**CDCPUB_10026** The process [process_name] with thread ID [thread_ID] encountered an error. Error code: [error_code]. Description: [error_description].

Explanation: The specified internal process encountered an error. This message provides information about the error.

User Response: To diagnose and correct the error, analyze the accompanying log messages. If the problem persists, contact Informatica Global Customer Support.

**CDCPUB_10027** One of the subordinate threads or processes for this process is no longer running.

Explanation: An internal process detected that its helper or subordinate process is not running.

User Response: Review the accompanying error messages from the helper or subordinate process. If the problem persists, contact Informatica Global Customer Support.

**CDCPUB_10028** The process [current_process_name] could not read or find the file [file_name].

Explanation: The specified internal process could not find or does not have access to the specified file.

User Response: Verify that the file exists and that read permissions have been granted on the file. If the problem persists, contact Informatica Global Customer Support.

**CDCPUB_10029** The process [current_process_name] found the directory name [directory_name] when expecting a file name.

Explanation: The internal process expected a file name but found the specified directory name instead.

User Response: Verify that the configuration properties that specify files and directories are set correctly. Also, verify that you have not inadvertently specified a directory name instead of a file name.

**CDCPUB_10030** The process [process_name] with thread ID [thread_ID] encountered a non-fatal error. Error description: [description].

Explanation: The specified process encountered a non-fatal error. The process should be able to ignore the error and continue running. The process will attempt to process the condition that caused the error again.
User Response: To diagnose and correct the error, analyze the accompanying log messages. If the problem persists, contact Informatica Global Customer Support.

**CDCPUB_10031** The process [process_name] with thread ID [thread_ID] will restart from the default position in the change stream. An explicit location is not required.

Explanation: This informational message reports that the specified process is restarting from the default location in the change data stream. The process cannot find or identify an explicit or logical restart point. The default location is typically the start or end of the change stream.

User Response: If you do want the process to restart from the default location, reconfigure the environment to restart from the location you want to use. If necessary, contact Informatica Global Customer Support.

**CDCPUB_10032** The process [process_name] with thread ID [thread_ID] will restart from the beginning or earliest available position in the change stream.

Explanation: This informational message reports that the specified process is restarting from the beginning of the change data stream or from the earliest available position in the change stream.

User Response: If you do not want to start from the beginning or earliest available position in the stream, reconfigure the environment to restart from a specific location. If necessary, contact Informatica Global Customer Support.

**CDCPUB_10033** The process [process_name] with thread ID [thread_ID] will restart from the end or latest available position in the stream.

Explanation: This informational message indicates that the specified process is restarting from the end of the change data stream or from the latest available position in the change stream.

User Response: If you do not want to restart from the end of or latest available position in the change stream, reconfigure the environment to restart from a specific location. If necessary, contact Informatica Global Customer Support.

**CDCPUB_10034** The process [process_name] with thread ID [thread_ID] will restart from the stream position identified by hex value [0xvalue].

Explanation: This informational message indicates that the specified process is restarting from the change data stream position that is represented by the hexadecimal value.

User Response: If you do not want to restart from this position, you can reconfigure the environment to restart from a specific location. If necessary, contact Informatica Global Customer Support.

**CDCPUB_10035** The process [process_name] was unable to stop processing on the path [path_name].

Explanation: The specified process could not stop processing data on the specified logical internal path. A path is the combination of internal processes responsible for moving data from a source to a target.

User Response: Typically, the process that owns the path retries the stop request.

**CDCPUB_10036** The process [process_name] is waiting for path [path_name] to stop processing data. Current wait duration: [milliseconds].

Explanation: This informational message reports that the process has been waiting the specified number of milliseconds for data processing on the specified logical internal path to stop. A path is the combination of internal processes responsible for moving data from a source to a target.
The process [process_name] was unable to shut down processing on the path [path_name].

Explanation: The process failed to shut down processing on the specified logical internal path. A path is the combination of internal processes responsible for moving data from a source to a target.

User Response: The process that owns the path typically retries the shutdown. If the problem persists, contact Informatica Global Customer Support.

The process [process_name] is waiting for the path [path_name] to shut down. Current wait duration: [milliseconds].

Explanation: The process has been waiting the specified number of milliseconds for processing on the logical internal path to shut down. Processing is still active. A path is the combination of internal processes responsible for moving data from a source to a target.

User Response: No response is required.

The process [process_name] with thread ID [thread_ID] is configured to use a pre-existing PowerExchange environment.

Explanation: This informational message indicates that the specified process is using an existing Informatica PowerExchange environment as the source of change data. You do not need to create a new PowerExchange environment as the source.

User Response: No response is required.

The process [process_name] could not interpret the restart configuration value [configuration_parameter_value]. The value is expected to start with one of the following prefixes: [prefixes].

Explanation: The process tried to use the specified restart configuration parameter value to determine the location in the change data stream from which to begin consuming data. However, the process could not interpret that value. The configuration value does not have a prefix or initial set of characters that match any of the expected characters.

User Response: Review the Informatica CDC Publisher documentation for the proper format for the configuration parameter value. If the problem persists, contact Informatica Global Customer Support.

The process [process_name] could not find or read the file [file_name] that is requested in the configuration parameter [parameter_name].

Explanation: The internal process could not find or does not have read permissions on the specified file. This message identifies the configuration parameter that specifies the file name.

User Response: Verify that the file exists and that read permissions have been granted on the file. Alternatively, edit the configuration parameter value to point to the correct file. If the problem persists, contact Informatica Global Customer Support.

The process [process_name] encountered an error while performing [action]. Error: [error_description].

Explanation: The specified internal process encountered an error while performing the specified action or step. The message includes a description of the error.

User Response: To diagnose and correct the error, analyze this message and accompanying log messages. If the problem persists, contact Informatica Global Customer Support.
The process [process_name] encountered an error while processing the configuration [parameter_name] with the value [parameter_value]. Error: [error_description].

Explanation: The specified internal process encountered an error while processing configuration information. This message reports the name and value of the configuration parameter that was being processed when the error occurred.

User Response: If the configuration parameter value was entered incorrectly, correct the value. If the problem persists, contact Informatica Global Customer Support.

The process [process_name] connected successfully to [entity].

Explanation: This informational message reports that the specified process successfully connected to an entity.

User Response: No response is required.

The process [process_name] was unable to create the directory [directory_name] requested in the configuration parameter [parameter_name].

Explanation: An internal process was unable to create the directory that is named in the specified configuration parameter. This message identifies the configuration parameter name and the directory name.

User Response: Perform one of the following actions:
- Correct the configuration value.
- Change the file system permissions to allow the process to create the directory.
- Create the directory manually, eliminating the need for the process to create it.
If the problem persists, contact Informatica Global Customer Support.

The process [process_name] was unable to create the directory [directory_name] requested in the configuration parameter [parameter_name]. Error: [errorDescription_or_message].

Explanation: The internal process was unable to create the directory that is identified in the specified configuration parameter. This message includes the directory name, parameter name, and error description or message.

User Response: Perform one of the following actions:
- Correct the configuration value if it is incorrect.
- Change the file system permissions to allow the process to create the directory.
- Create the directory manually, eliminating the need for the process to create it.
If the problem persists, Contact Informatica Global Customer Support for assistance.
CDCPUB_10048  CDC Publisher release [release] build number [build_number] dated [date].
Explanation: This informational message reports the CDC Publisher release, build number, and build date.
User Response: No response is required.

CDCPUB_10050  The environment [environment_item] has the value [value].
Explanation: This informational message reports environment variables, library paths, or other information about the environment where the CDC Publisher process is running.
User Response: No response is required.

CDCPUB_10051  The process [process_name] successfully loaded the external library [library_name].
Explanation: This informational message reports that the process successfully loaded the specified external library.
User Response: No response is required.

CDCPUB_10052  The process [process_name] was unable to load the external library [library_name]. Error: [errorDescription_or_message].
Explanation: The specified internal process was unable to load the specified external library. This message provides the error message or description.
User Response: Perform one of the following actions:
- Add or download the library if it is missing.
- Correct environment settings to include the library in the system path.
If the problem persists, contact Informatica Global Customer Support.

CDCPUB_10053  The configuration parameter [parameter_name] has the value [value] which is outside the allowable range of [lowest_value] to [highest_value].
Explanation: The specified configuration parameter is set to a value that is not within the range of allowable values that is identified in this message.
User Response: To correct the error, change the configuration parameter value to a value that is within the range of allowable values.

CDCPUB_10054  The configuration parameter [parameter_name] has the invalid value [value]. Allowable values are [allowable_values].
Explanation: The specified configuration parameter is set to a value that is not allowed or recognized.
User Response: To correct the error, change the configuration parameter value to one of the allowed values that are identified in this message.

CDCPUB_10055  The configuration parameter [parameter_name] has a null or empty value. This parameter requires a valid value.
Explanation: The specified configuration parameter has a null or empty value. It must be set to a valid value.
User Response: To correct the error, enter a valid value for the configuration parameter.

CDCPUB_10056  The CDC Publisher could not find the required configuration parameter [parameter_name].
Explanation: The CDC Publisher could not find the specified configuration parameter, which is required.
User Response: Verify that the configuration parameter is appropriate for your environment and then add it to your configuration. If the parameter is not appropriate, the CDC Publisher might still expect it because another related parameter is incorrectly set. In this case, review the other configuration parameter settings to determine if any of them are inadvertently requiring the specified parameter to be configured. If the problem persists, contact Informatica Global Customer Support.

**CDCPUB_10057**

One or more configuration parameters in the logical group named \[group_name\] have a value that is in error. Error: \[description_or_message\].

Explanation: The specified logical group contains one or more configuration parameters that have invalid values. The message includes the error description or message.

User Response: Review the information in this message and accompanying messages to determine the configuration parameters that have the invalid values and the corrective actions to take.

**CDCPUB_10058**

The process \[process_name\] found the file \[file_name\] when expecting a directory name.

Explanation: The internal process found the specified file name but expected to find a directory name.

User Response: Verify that the configuration properties that specify files and directories are set correctly. Make sure that you have not inadvertently specified a file name instead of a directory name.

**CDCPUB_10059**

The configuration property \[property_name\] has the value \[value\].

Explanation: This informational message reports a configuration property value that is in use by the CDC Publisher process.

User Response: No response is required.

**CDCPUB_10060**

The server is listening on port \[port_number\].

Explanation: This informational message reports the port number that the CDC Publisher server is using.

User Response: No response is required.

**CDCPUB_10061**

The configuration parameter \[parameter_name\] has been deprecated. Use alternate parameter \[parameter_description\].

Explanation: This informational message reports that the specified configuration parameter has been deprecated and identifies the parameter that replaces it, if available.

User Response: No response is required.

**CDCPUB_10062**

The CDC Publisher detected that the value of the configuration parameter \[parameter_name\] is not valid because it is not a string representation of a hexadecimal value.

Explanation: The CDC Publisher detected that the specified configuration parameter has an invalid value. The value should be a string representation of a hexadecimal value.

User Response: For the specified configuration parameter, enter a valid hexadecimal string that consists of an even number of characters, which can include the alphanumeric characters 0-9 and A-F. If the problem persists, contact Informatica Global Customer Support.

**CDCPUB_10063**

Executing product release \[release\] build number \[build_number\] dated \[date\].

Explanation: This informational message reports the product release, build number, and build date for the product that is running.

User Response: No response is required.
The server is listening on host [host_IP_address] and port [port_number].

Explanation: This informational message identifies the host IP address and port number that the CDC Publisher server is using.

User Response: No response is required.

No valid license found.

Explanation: The CDC Publisher found no license for the product or one of its features.

User Response: Check the system log for additional messages that provide details about the license error. If you need assistance, contact Informatica Global Customer Support.

TRACE: [trace_information]

Explanation: This error message provides trace-level information about processing at the time of an error.

User Response: To diagnose and correct the error, analyze the accompanying log messages. If the problem persists, provide this trace information and any related messages to Informatica Global Customer Support for further analysis.

File [file_name]:line

Explanation: This informational message reports the contents of the specified file. One message is produced for each line in the file.

User Response: No response is required.

The configuration property [property_name] in group [group_name] is not recognized. Please verify that the property name is correct.

Explanation: The specified configuration parameter is not recognized.

User Response: Verify that the configuration property is spelled correctly and in the correct group of properties.

The CDC Publisher requested the main process to force a checkpoint to the target. With a timeout value of [number_of_milliseconds] milliseconds.

Explanation: This informational message reports that the CDC Publisher requested its main process to write a checkpoint to the target. The checkpoint can be used to validate that all change data has been consumed up to this point. The request will time out with an error if it is not successful within the timeout interval, which is specified in milliseconds.

User Response: No response is required.

The CDC Publisher request to force a checkpoint to the target completed successfully after [number_of_milliseconds] milliseconds.

Explanation: This informational message indicates that a CDC Publisher request to force a checkpoint to the target completed successfully. The request ran for the reported number of milliseconds.

User Response: No response is required.
The main CDC Publisher process was unable to force a checkpoint to the target. Error: [description_or_message].

Explanation: The CDC Publisher main process was unable to complete the request to force a checkpoint and validate that all change data has been sent to the target. This error can occur if the target system is unavailable or if the time required to send all data to the target exceeds the amount of time the caller can wait for completion.

User Response: This message does not indicate that data loss occurred. Data that has not yet reached the target will be sent to the target the next time the process restarts. If this message is issued repeatedly, it might indicate a slow system or a timeout configuration that is too restrictive. Analyze the accompanying log messages to diagnose and correct the error. If the problem persists, contact Informatica Global Customer Support.

The process [process_name] has the following statistics:

Explanation: This informational message introduces CDC Publisher statistics, which are presented in subsequent messages.

User Response: No response is required.

stat_owner : stat_name : stat_value

Explanation: This informational message reports, for a given statistic, the process that owns or created the statistic and the statistic name and value.

User Response: No response is required.

The CDC Publisher request to force a checkpoint succeeded. Checkpoint label [owner_name] and sequence value [sequence_value].

Explanation: This informational message reports that a CDC Publisher request to force a checkpoint succeeded. The message includes the sequence value from the checkpoint and the owner of the sequence value. A sequence value indicates the position in the change data stream up to which change data has been processed and applied to the target.

User Response: No response is required.

The CDC Publisher request to force a checkpoint succeeded. Checkpoint sequence value [sequence_value].

Explanation: This informational message reports that a CDC Publisher request to force a checkpoint succeeded. The message includes the sequence value from the checkpoint. A sequence value indicates the position in the change data stream up to which change data has been processed and applied to the target.

User Response: No response is required.

Other checkpoint information [checkpoint_information].

Explanation: This informational message provides additional information about a checkpoint. To determine the checkpoint label (owner name) and sequence in the change stream, see the preceding checkpoint messages.

User Response: No response is required.
The process [process_name] is waiting for verification that all work has been sent to the target. Current wait duration: [milliseconds].

Explanation: The CDC Publisher process is waiting for its request to verify that all data has been sent to the target to complete. This message reports the current wait time and is issued periodically when the verification takes longer than a few seconds.

User Response: No response is required.

Memory :[details].

Explanation: This informational message reports a memory consumption attribute or statistic from the CDC Publisher. Multiple instances of this message can occur, each reporting a different element.

User Response: No response is required.

Processing a request to clear the cache for Formatter [formatter_name].

Explanation: This informational message reports that a request to clear the cache for the specified Formatter is being processed. The message includes the logical name of the Formatter.

User Response: No response is required.

Processing a request to clear the cache of Formatter [formatter_name] for table [table_name].

Explanation: This informational message reports that a request to clear a table entry from the cache of the specified Formatter is being processed. The message includes the logical name of the Formatter and the name of the table.

User Response: No response is required.

Processing a request to print the cache entries for Formatter [formatter_name].

Explanation: This informational message reports that a request to print the contents of the cache for the specified Formatter is being processed. The message includes the logical name of the Formatter.

User Response: No response is required.

Processing a request to print the a table entry in the cache of Formatter [formatter_name] for table [table_name].

Explanation: This informational message reports that a request to print a table entry that is in the cache of the specified Formatter is being processed. The message includes the logical name of the Formatter and the name of the table.

User Response: No response is required.

Processing a request to load or reload the cache of the Formatter [formatter_name].

Explanation: This informational message reports that a request to load or reload cache entries for the specified Formatter is being processed. Only new or missing entries will be created. Typically, existing cache entries are not overwritten. To replace an entry, you must clear it from cache first.

User Response: No response is required.
The request to load or reload the formatter cache for Formatter [formatter_name] failed. The load process ended prematurely.

Explanation: A request to load or reload the cache for the specified Formatter failed. Additional messages that describe the error in detail should have preceded this message.

User Response: For more information about the error, see the preceding messages. After you diagnose the cause of the error, try to correct the problem and restart the process. If the problem persists, contact Informatica Global Customer Support.

The request to load or reload the cache for Formatter [formatter_name] is waiting for extraction maps. Current wait duration: [milliseconds].

Explanation: This informational message reports the number of milliseconds that the specified Formatter has been waiting to receive the extraction maps that were read from PowerExchange so that it can load or reload its cache.

User Response: No response is required.

The request to load or reload the cache for Formatter [formatter_name] failed. The process failed to complete within a reasonable amount of time.

Explanation: A request to load the cache of the specified Formatter did not finish within the maximum time allowed. As a result, the process was terminated before the cache was completely loaded.

User Response: If the cache entries that you need were loaded, you can ignore this error. Otherwise, run the request to load the cache again. If the problem persists, contact Informatica Global Customer Support.

A request to perform JVM garbage collection processing was executed by [process_name] with thread ID [thread_ID].

Explanation: This informational message reports that JVM garbage collection processing was successfully executed by an internal process on the specified thread.

User Response: No response is required.

The process [process_name] found configuration property [property_name] in both file [file_name1] and file [file_name2].

Explanation: This informational message reports that the specified process found multiple configuration files with the same configuration property.

User Response: Informatica recommends that a configuration property be specified in only one configuration file in your environment.

The process [process_name] found configuration property [property_name] in file [file_name1] and file [file_name2] with conflicting values.

Explanation: The process found that two configuration files specify the same configuration property but with different values. The process cannot determine the correct value to use.

User Response: Edit the configuration files to ensure that the property is specified in only one of the configuration files with the correct value. If the problem persists, contact Informatica Global Customer Support.
CDCPUB_11024  The process [process_name] found no configuration files that have suffix [file_suffix] and contain header text [file_header] in the directory [directory_name].

Explanation:  The process found no configuration files that have the required file-name suffix and header content in the specified directory. As a result, no configuration information is available.

User Response:  Verify that the requested directory name is correct and that this directory contains configuration files that have the expected file-name suffix and header content. If the problem persists, contact Informatica Global Customer Support.

CDCPUB_11025  The process [process_name] is reading configuration properties file [file_name].

Explanation:  This informational message reports the name of the configuration file that the specified process is reading.

User Response:  No response is required.

CDCPUB_11026  Processing to load or reload the cache of the Formatter [formatter_name] completed. number_of_items items were loaded.

Explanation:  This informational message reports that a request to load or reload cache entries for the specified Formatter successfully completed. The specified number of items were loaded.

User Response:  No response is required.

CDCPUB_11027  The requested instance directory [directory_name] was not found.

Explanation:  The specified instance directory name was not found or is not a directory.

User Response:  Verify that the correct instance directory name was provided. Make sure that you have not inadvertently specified a file name instead of a directory name.

CDCPUB_11028  Using instance directory [directory_name].

Explanation:  This informational message reports the instance directory that the product is being used for configuration, logging, reporting, and other files.

User Response:  No response is required.

CDCPUB_13001  Internal Logic Error: The Capture Consumer on thread [thread_name] could not find a cached schema for the object ID [object_ID] when processing a DML event at the change stream sequence position [sequence_position].

Explanation:  This error message is generated by the internal Capture Consumer process that is responsible for consuming or capturing change data. The process caches information about the schema or definition of the object for which change data is being captured or consumed. However, the process could not retrieve the object schema or definition from the cache because of an internal logic error.

User Response:  Restart the process to re-create the cached entry. If the problem persists, contact Informatica Global Customer Support.

CDCPUB_13002  The Capture Consumer [process_name] with the thread ID [thread_ID] is starting.

Explanation:  This informational message reports that the internal Capture Consumer process is starting.

User Response:  No response is required.
**CDCPUB_13003** The Capture Consumer process_name with the thread ID [thread_ID] shut down successfully.

Explanation: This informational message reports that the internal Capture Consumer process shut down successfully.

User Response: No response is required.

**CDCPUB_13004** The Capture Consumer [process_name] with the thread ID [thread_ID] stopped running because of a fatal error.

Explanation: The internal Capture Consumer process that is responsible for consuming or capturing change data stopped running because of a fatal error.

User Response: To diagnose and correct the error, analyze the accompanying log messages. If the problem persists, contact Informatica Global Customer Support.


Explanation: The internal Capture Consumer process that is responsible for consuming or capturing change data ended because of a fatal error. This message provides the error code and description.

User Response: To diagnose and correct the error, analyze the accompanying log messages. If the problem persists, contact Informatica Global Customer Support.

**CDCPUB_13006** The Capture Consumer [process_name] with thread ID [thread_ID] encountered an I/O error. Resetting the connection to the Change Data Capture stream. The last stream position was [position].

Explanation: The internal Capture Consumer process that is responsible for consuming or capturing change data has reset its connection to the entity that creates the change data capture stream. This message might be issued after a brief network outage or on a periodic basis to ensure that an active connection exists. The message reports the last position in the change stream that was processed prior to the error.

User Response: If this message is issued repeatedly, check if a network outage exists and if the entity responsible for generating the change data capture stream is operating.

**CDCPUB_13007** The Capture Consumer [process_name] with the thread ID [thread_ID] is waiting for the change data capture stream to be re-established. Current wait time: [milliseconds].

Explanation: The internal Capture Consumer process that is responsible for consuming or capturing change data is in a wait state because the entity responsible for generating the change stream is unavailable. This informational message reports the amount of time, in milliseconds, that this process has been waiting.

User Response: No response is required.

**CDCPUB_13008** The Capture Consumer [process_name] with the thread ID [thread_ID] is waiting for data from the change data capture stream. Current wait duration: [milliseconds].

Explanation: The internal Capture Consumer process that is responsible for consuming or capturing change data is in a wait state because it is not receiving change data from the entity that is responsible for generating the change stream. This informational message reports the amount of time, in milliseconds, that this process has been waiting.

User Response: No response is required.
The Capture Consumer [process_name] with the thread ID [thread_ID] is waiting to write data from the change data capture stream because the internal InputStream is at capacity. Current wait duration: [milliseconds].

Explanation: The internal Capture Consumer process that is responsible for consuming or capturing change data is in a wait state because the internal buffer that holds the change data is full. This informational message reports the amount of time, in milliseconds, that this process has been waiting. Typically, the wait time is a short. As change data is processed, items are removed from the buffer, allowing this process to continue.

User Response: If this message is issued repeatedly, the change data is probably arriving at a rate faster than the rate at which it is being processed. To increase the processing rate, increase the buffer sizes or number of helper threads.

The Capture Consumer [process_name] with the thread ID [thread_ID] is shutting down.

Explanation: This informational message reports that the internal Capture Consumer process that is responsible for consuming or capturing change data is shutting down.

User Response: No response is required. If necessary, contact Informatica Global Customer Support.

The Capture Filter [process_name] with the thread ID [thread_ID] is starting.

Explanation: This informational message reports that the specified internal Capture Filter process, which filters the change data to be captured or consumed, is starting.

User Response: No response is required.

The Capture Filter [process_name] with the thread ID [thread_ID] shut down successfully.

Explanation: This informational message reports that the internal Capture Filter process that is responsible for filtering the change data to be captured or consumed has successfully shut down.

User Response: No response is required.

The Capture Filter [process_name] with the thread ID [thread_ID] stopped running because of a fatal error.

Explanation: The internal Capture Filter process that is responsible for filtering change data to be captured or consumed has stopped because of a fatal error.

User Response: To diagnose and correct the error, analyze the accompanying log messages. If the problem persists, contact Informatica Global Customer Support.

The Capture Filter [process_name] with the thread ID [thread_ID] stopped because of a fatal error. Error code: [error_code]. Error description: [error_description].

Explanation: The internal Capture Filter process that is responsible for filtering change data stopped because of a fatal error. This message provides the error code and a description of the error.

User Response: To diagnose and correct the error, analyze the error information in this message and any accompanying log messages. If the problem persists, contact Informatica Global Customer Support.

The process process_name with the thread ID thread_ID is waiting for incoming data. Current wait duration: [milliseconds].

Explanation: The internal process that is responsible for change data capture is sleeping because it has no input data to process. This situation is typical for an idle system.
User Response: No response is required.

CDCPUB_13016 The CDC Publisher Extractor [process_name] requested a shutdown of its helper thread [thread_name].

Explanation: This informational message reports that the CDC Publisher Extractor process has requested a helper or subordinate thread to shut down.

User Response: No response is required.

CDCPUB_13017 The CDC Publisher Extractor [process_name] was unable to shut down its helper thread [thread_name].

Explanation: This informational message reports that the CDC Publisher Extractor process was unable to shut down the specified helper or subordinate thread.

User Response: No response is required.

CDCPUB_13018 The CDC Publisher Extractor [process_name] is waiting for its helper thread [thread_name] to shut down. Current wait duration: milliseconds.

Explanation: The CDC Publisher Extractor process is waiting for a helper or subordinate thread to shut down. This informational message reports the amount of time, in milliseconds, that this process has been waiting.

User Response: No response is required.


Explanation: The internal Capture Consumer process that is responsible for consuming or capturing change data is waiting for the PowerExchange Capture Service interface to shut down. This informational message reports the amount of time, in milliseconds, that this process has been waiting for the PowerExchange Capture Service to shut down.

User Response: No response is required.

CDCPUB_13020 The Capture Consumer [process_name] with the thread ID [thread_ID] has issued the command [command] to the PowerExchange Capture Service.

Explanation: This informational message reports that the internal Capture Consumer process that is responsible for consuming or capturing change data issued the specified command to the PowerExchange Capture Service interface.

User Response: No response is required.

CDCPUB_13021 The Capture Consumer [process_name] with the thread ID [thread_ID] stopped waiting for the PowerExchange Capture Service to shut down after wait time milliseconds.

Explanation: This informational message reports that the internal Capture Consumer process that is responsible for consuming or capturing change data stopped waiting for the PowerExchange Capture Service interface to shut down after the specified number of milliseconds elapsed. The Capture Consumer process will shut down.

User Response: No response is required.
CDCPUB_13022  The Capture Consumer [process_name] with the thread ID [thread_ID] is resetting the connection to the change data capture stream because of a lack of activity. The last position in the change stream was [current_position].

Explanation: The internal Capture Consumer process that is responsible for consuming or capturing change data reset its connection to the change data capture provider after a period of no change activity. This informational message reports the current position or sequence in the change stream. The message might be issued periodically when the capture process is idle. The connection is reset to verify that the change data capture provider is still active.

User Response: No response is required.

CDCPUB_13023  The Capture Consumer [process_name] with the thread ID [thread_ID] is starting the PowerExchange Capture Service.

Explanation: This informational message reports that internal Capture Consumer process that is responsible for consuming or capturing change data is starting the PowerExchange Capture Service interface.

User Response: No response is required.

CDCPUB_13024  The process [process_name] could not interpret the configured restart value [restart_value] correctly. The value is not a numeric value or hex string.

Explanation: A configured setting identifies the location in the change data stream where the process should resume consuming change data. However, this error message indicates that the configured restart value is not a valid numeric value or hexadecimal string.

User Response: Review the Informatica product documentation to determine the proper format for specifying a restart point. If the problem persists, contact Informatica Global Customer Support.

CDCPUB_13025  The process [process_name] with the thread ID [thread_ID] is waiting until more space is available on its internal event queue. Current wait duration: [milliseconds].

Explanation: This informational message reports that the internal process that is responsible for processing change data is idle because it does not have adequate space available to process more data. The process has waited the specified number of milliseconds for more space. This condition typically occurs when the process cannot find an available helper thread for processing event data, because all helper threads are busy.

User Response: No response is required.

CDCPUB_13026  The process [process_name] with the thread ID [thread_ID] is waiting until more space is available on its output queue. Current wait duration: [milliseconds].

Explanation: This informational message reports that the internal process that is responsible for processing change data is idle because it does not have adequate space available to write results to an output queue. The process has waited the specified number of milliseconds to write results. This condition typically occurs when a downstream process is not consuming events as quickly as the process provides them.

User Response: No response is required.
The process [process_name] with the thread ID [thread_ID] exceeded the maximum wait time for writing data from the capture stream because the internal InputStream is at capacity. Current wait duration: [seconds_elapsed] seconds. Maximum wait time allowed: [maximum_seconds] seconds.

Explanation: The specified process has reached the maximum amount of time, in seconds, to wait on a full input buffer before treating the full buffer condition as an error. The full buffer condition typically occurs when other processes are unable to consume data from the buffer.

User Response: Review the system log to determine why downstream processes are not processing data. If the problem persists, contact Informatica Global Customer Support.

The Capture Consumer process [process_name] with the thread ID [thread_ID] failed to start a change data stream. Return code: [return_code].

Explanation: The specified process failed to start a change data stream. The message provides the non-zero return code for the error.

User Response: To diagnose the error, review the accompanying messages in the system log. If the problem persists, contact Informatica Global Customer Support.

The Capture Consumer process [process_name] with the thread ID [thread_ID] found that the underlying change data stream stopped. Return code: [return_code].

Explanation: The specified process detected that the change data stream stopped unexpectedly. The message includes the non-zero return code for the error.

User Response: To diagnose the error, review the accompanying messages in the system log. If the problem persists, contact Informatica Global Customer Support.

The Capture Consumer process [process_name] with the thread ID [thread_ID] stopped the change data stream because of a shutdown request. Return code: [return_code].

Explanation: After the specified process received a shutdown request, it stopped the change data stream. The stream ended with specified return code.

User Response: No response is required.

The Capture Consumer process [process_name] with the thread ID [thread_ID] successfully shut down the change data stream but the stream ended with the error: [error_description].

Explanation: The specified process successfully shut down the change data stream, even though the reported error occurred during the process. Because the error did not prevent the shutdown from completing successfully, you can ignore the error.

User Response: Ignore the error. If it occurs again repeatedly, contact Informatica Global Customer Support.

The Extractor [logical_extractor_name] processed the request [input_request] to generate an internal restart value. Generated restart value: [restart_value].

Explanation: This informational message reports the internal restart value that Extractor process generated in response to the specified request.

User Response: No response is required. However, if the result did not match the input request or if you do not expect a restart, contact Informatica Global Customer Support.
CDCPUB_14001  The Formatter [process_name] is waiting for its helper thread [thread_name] to shut down. Current wait duration: [milliseconds].

Explanation: The internal Formatter process that is responsible for formatting change data is waiting for its helper or subordinate thread to shut down. This informational message includes the current elapsed wait time, in milliseconds.

User Response: No response is required.

CDCPUB_14002  Internal logic error: The formatter helper thread [thread_name] could not find a schema cached for the object ID [object_ID] when processing a DML event at the change stream sequence location [sequence_location].

Explanation: This error message is generated by the internal Formatter process that is responsible for formatting change data. The process caches information about the schema of an object for which it consumes change data. This message indicates that an internal logic error prevented the Formatter process from retrieving the schema for the specified object from an internal cache when processing a DML change operation. The message includes the object ID and the current sequence location within the change stream of the DML operation.

User Response: To re-create the cached entry, restart the Formatter process. If the problem persists, contact Informatica Global Customer Support.

CDCPUB_14003  The Formatter helper [process_name] encountered an error and ended while formatting the event at the change stream location [sequence_location]. Error description: [error_description].

Explanation: The internal Formatter process that is responsible for formatting change data encountered an error and ended unexpectedly. This message includes a brief description of the error and the current sequence location within the change data stream.

User Response: To diagnose and correct the error, analyze the accompanying log messages. If the problem persists, contact Informatica Global Customer Support.

CDCPUB_14004  The Formatter [process_name] encountered an error while attempting to create a schema cache. Error description: [error_description].

Explanation: The internal Formatter process that is responsible for formatting change data was unable to create an internal cache to hold schema information.

User Response: To diagnose and correct the error, analyze the accompanying log messages. If the problem persists, contact Informatica Global Customer Support.

CDCPUB_14005  The Formatter [process_name] encountered an error while attempting to create an Avro schema cache. Error description: [error_description].

Explanation: The internal Formatter process that is responsible for formatting change data into Avro messages was unable to create an internal cache to hold Avro schema information.

User Response: To diagnose and correct the error, analyze the accompanying log messages. If the problem persists, contact Informatica Global Customer Support.
CDCPUB_14006  The table [table_name] contains characters that are not valid for an Avro schema. Avro field names must begin with a letter and include only the digits 0-9, letters A-Z and a-z, and underscore (_) character. Consider changing the current replacement character [replacement_character].

Explanation: The table name is used to create an Avro field name. Avro field names must begin with an alphabetic character A-Z or a-z and can contain the digits 0-9, letters A-Z and a-z, and the underscore (_) character. If a replacement character is defined, it will replace any non-conforming characters in the table name before the name is passed to Avro.

User Response: Verify that you defined a replacement character that can be used to replace any special characters in the table name. If you suspect that the current replacement character is not supported by Avro, try changing the replacement character. If the problem persists, contact Informatica Global Customer Support.

CDCPUB_14007  The column [column_name] in table [table_name] contains characters that are not valid for an Avro schema. Avro field names must begin with a letter and include only the digits 0-9, letters A-Z and a-z, and underscore (_) character. Consider changing the current replacement character [replacement_character].

Explanation: The column name is used to create an Avro field name. Avro field names must begin with an alphabetic character A-Z or a-z and can contain the digits 0-9, letters A-Z and a-z, and the underscore (_) character. If a replacement value is defined, it will replace any non-conforming characters in the column name before the name is passed to Avro.

User Response: Verify that you defined a replacement character that can be used to replace any special characters in the column name. If you suspect that the current replacement character is not supported by Avro, try changing the replacement character. If the problem persists, contact Informatica Global Customer Support.

CDCPUB_14008  The process [process_name] failed to find an entry for the table [table_name]. The process will retry the lookup by using the following technique: [retry_technique].

Explanation: The process could not find an entry for specified table name. The process is retrying the table lookup by using a different technique.

User Response: No response is required.

CDCPUB_14009  All attempts by process [process_name] to find any entry for [table_name] failed.

Explanation: The process received a request to find the internal entities or entries for the specified table name, but all attempts to find these items failed. Either the table name was entered incorrectly or the system is not configured to process this table.

User Response: Verify that you provided the correct table name. Also consider changing the table name to avoid any filtering or modifications that the system is configured to perform on a source table name. If the problem persists, contact Informatica Global Customer Support.

CDCPUB_14010  When generating a wrapped Avro schema, the process [process_name] found an unsupported Avro configuration. To generate a wrapped Avro schema, the outer Avro message must support or be the same as the inner Avro message encoding. Current outer encoding [outer_encoding] does not support the current inner encoding [inner_encoding].

Explanation: The encoding settings for messages that use a wrapped Avro schema are not valid. The encoding of the outer “wrapper” message must be binary, or the encoding of the inner “wrapped” message must match the encoding of the outer message.
User Response: Correct one or both of the Avro message encoding settings so that they match or the outer message encoding is binary. If the problem persists, contact Informatica Global Customer Support.

**CDCPUB_14011** When generating a wrapped Avro schema, the process [process_name] determined that the configured encoding of the inner 'wrapped' message is not supported. Inner message encoding [requested_inner_encoding] was replaced with [valid_encoding].

Explanation: When generating a wrapped Avro schema, the process determined that the encoding of the inner "wrapped" Avro message is not supported. As a result, the process replaced the requested invalid encoding with the specified valid encoding.

User Response: No response is required

**CDCPUB_15001** The Connector [process_name] is waiting for its helper thread [thread_name] to shut down. Current wait duration: [milliseconds].

Explanation: The internal Connector process, which is responsible for connecting to and sending messages to an external entity, is waiting for a helper or subordinate thread to shut down. This informational message includes the amount of time, in milliseconds, that this process has been waiting.

User Response: No response is required.

**CDCPUB_15002** The Connector helper [process_name] encountered an error when sending the event at the change stream location [sequence_location]. Error description: [error_description].

Explanation: The internal Connector process, which is responsible for connecting to and sending messages to an external entity, encountered an error and ended when processing the event at the specified change stream location.

User Response: To diagnose and correct the error, analyze the accompanying log messages. If the problem persists, contact Informatica Global Customer Support.

**CDCPUB_15003** The Connector [process_name] encountered an error while trying to connect to the target. Error description: [error_description].

Explanation: The internal Connector process was unable to establish a connection to an external entity. This message includes a brief description of the error.

User Response: Verify that the entity to which the process is attempting to connect is available. If the problem persists, Contact Informatica Global Customer Support.

**CDCPUB_15004** The Connector [process_name] will write output to the file [file_name].

Explanation: This informational message reports that the internal Connector process is configured to write message output to the specified file.

User Response: No response is required.
CDCPUB_15005  The Connector helper [helper_process_name] with the thread ID [thread_ID] is waiting for a request to send data to the target. Current wait duration: [milliseconds].

Explanation: The internal Connector helper process, which is responsible for sending data to the target, is waiting for permission to proceed. In some configurations, the order in which messages are sent to the target entity is strictly enforced. This informational message might be issued when a helper process is ready to send its message but must wait for its turn. If this message is issued frequently or consistently, the number of helpers configured might be greater than the number needed for the current volume of data. When too many threads exist, they might wait on one another.

User Response: No response is required.

CDCPUB_15006  The process [process_name] with the thread ID [thread_ID] could not find any Connectors defined in the current configuration. No paths will be created to send change data.

Explanation: No Connectors are defined in the current configuration. As a result, the specified process cannot send change data to any external entity on the logical path. A logical path is the combination of internal Extractor, Formatter, and Connector processes that are responsible for moving captured source data to a target entity.

User Response: Typically, this informational message is issued if the configuration is incomplete or contains an error. If you need assistance, contact Informatica Global Customer Support.

CDCPUB_15007  The Process [process_name] with the thread ID [thread_ID] determined that the Connector [connector_name] references the CDC Publisher Extractor name [logical_extractor_name], but that name is not in the configuration parameter [parameter_name].

Explanation: The process determined that the specified Connector references an Extractor name that is not defined in the Connector configuration. This error message reports the Connector name, the Extractor name that cannot be found, and the Connector configuration parameter for specifying the Extractor name.

User Response: In the Connector configuration parameter, correct the Extractor name or add the name of an existing Extractor process.

CDCPUB_15008  The process [process_name] with the thread ID [thread_ID] found that the Connector [connector_name] references the filter name [filter_name], but that Filter name is not found in the configuration parameter [parameter_name].

Explanation: The process determined that the specified Connector references a filter name that is not defined in the Connector configuration. This error message includes the Connector name, the name of the filter that cannot be found, and the name of the Connector configuration parameter for specifying the filter name.

User Response: In the Connector configuration parameter, correct the correct filter name or add the name of an existing filter.

CDCPUB_15009  The process [process_name] with the thread ID [thread_ID] found that the Connector [connector_name] references the Formatter name [formatter_name], but that Formatter name is not found in the configuration parameter [parameter_name].

Explanation: The process determined that the specified Connector references a Formatter name that is not defined in the Connector configuration. This error message includes the Connector name, the name of the Formatter that was not found, and the name of the Connector configuration parameter for specifying the Formatter name.
User Response: In the Connector configuration parameter, correct the Formatter name or add the name of an existing Formatter.

**CDCPUB_15010** The process [process_name] with the thread ID [thread_ID] found no Connectors defined in the current configuration. The process expected to see the Connector names listed in the configuration parameter [parameter_name].

Explanation: The process could not find any Connector names in the specified configuration parameter. The parameter should list all Connectors. As a result, no data will be sent to any external entity.

User Response: This error message is typically produced if the configuration is incomplete or contains an error. If the problem persists, contact Informatica Global Customer Support.

**CDCPUB_15011** The process [process_name] with the thread ID [thread_ID] found that the Connector named [connector_name] has incomplete or inaccurate configuration information. The CDC Publisher will not use this Connector.

Explanation: This informational message reports that the process found an inaccurate or incomplete definition of the specified Connector in the configuration. The process will ignore this Connector. If complete and accurate information is defined for other Connectors in the configuration, one of these Connectors can be used.

User Response: If the configuration is complete, check if the configuration contains an error. If you need assistance, contact Informatica Global Customer Support.

**CDCPUB_15012** The process [process_name] with the thread ID [thread_ID] did not find the logical name [logical_name] in the configuration parameter [parameter_name].

Explanation: When examining configurations, the internal process expected to find the specified configuration parameter with the logical name of the entity for which configuration information is being processed. However, the configuration parameter was not found or did not contain the expected logical name.

User Response: Add the missing configuration parameter with the logical name to the configuration. If the problem persists, contact Informatica Global Customer Support for assistance.

**CDCPUB_15013** The process [process_name] with the thread ID [thread_ID] examined configurations for the logical name [logical_name_of_entity] but did not find this name in the configuration parameter [parameter_name].

Explanation: The internal process examined configurations for the specified logical name of a entity but did not find the logical name in the specified configuration parameter as expected.

User Response: Specify the logical name in the configuration parameter. If the problem persists, contact Informatica Global Customer Support.

**CDCPUB_15014** The Connector [process_name] will write checkpoint information to the file [file_name].

Explanation: The internal helper process that is responsible for sending messages to an external entity keeps a record of its progress, called a checkpoint. This informational message reports that the checkpoint information will be written to the specified file.

User Response: No response is required.
CDCPUB_15015 The Connector [process_name] will read parameter information from the file [file_name].

Explanation: The specified internal helper process that is responsible for sending messages to an external entity normally receives configuration properties from a parent entity or process. The informational message indicates that the process is getting some configuration properties directly from the specified file. This behavior is normal for configurations for which connectivity parameters can be in a file, such as configurations with Kafka targets.

User Response: No response is required.

CDCPUB_15016 The process [process_name] was unable to validate the Kafka partition ID [partition_ID] for the topic [topic_name]. Error: [error_description].

Explanation: The internal process that is responsible for Kafka handling found that the specified Kafka partition ID that is assigned to a message is not valid for the configured Kafka topic.

User Response: Change the Kafka topic definition to support the partition ID, or change the partition configuration to match the existing Kafka topic definition. Also, to diagnose and correct the error, analyze the accompanying log messages. If the problem persists, contact Informatica Global Customer Support.

CDCPUB_15017 The process [process_name] was processing an event with these details when an error occurred: [event_information].

Explanation: The internal process that is responsible for Kafka communication encountered an error. This message reports the contents of the event or message that was sent to Kafka.

User Response: This message typically accompanies additional messages that describe the error. To diagnose and correct the error, analyze the accompanying log messages. If the problem persists, contact Informatica Global Customer Support.

CDCPUB_15018 The process [process_name] encountered an error while sending a message to Kafka. Error: [error_description].

Explanation: The internal process that is responsible for Kafka communication encountered the specified error while sending data to Kafka.

User Response: After you review the error description, determine if you need to check the availability or configuration of the target Kafka environment or make changes to the CDC Publisher configuration. Also, analyze the accompanying log messages for additional diagnostic information. If the problem persists, contact Informatica Global Customer Support.

CDCPUB_15019 The process [process_name] received an [Kafka_exception_type] exception while sending a message to Kafka for the event [event_information]. Error: [error_description].

Explanation: The specified internal helper process that is responsible for sending messages to Kafka encountered the specified error. This message includes the exception and event information that was returned from Kafka.

User Response: After you review the error information, determine if you need to check the configuration of the target Kafka environment or make changes to the CDC Publisher configuration. Also, analyze the accompanying log messages for additional diagnostic information. If the problem persists, contact Informatica Global Customer Support.
The process [process_name] is trying to resend the previous message to Kafka because of an error. Error: [error_description].

Explanation: This informational message reports that the internal process that is responsible for sending messages to Kafka encountered the specified error. The process will retry sending the message until either the send operation is successful or the maximum number of retries is exceeded.

User Response: No response is required.

The process [process_name] was unable to send the previous message to Kafka after [number_of_attempts] attempts because of the error: [error_description].

Explanation: The internal process that is responsible for Kafka communication encountered the specified error while trying to resend a message with data to Kafka. After the maximum number of retries was exceeded, the process ended. This message includes the error description and number of retry attempts.

User Response: After you review the error description, determine if the error is related to the availability or configuration of the Kafka target environment. Edit the configuration if necessary. Also, analyze the accompanying log messages to diagnose the error. If the problem persists, contact Informatica Global Customer Support.

The process [process_name] received the following Kafka producer properties.

Explanation: This informational message introduces the list of Kafka properties that the internal process that is responsible for Kafka communication uses.

User Response: No response is required.

Kafka property key [property_name] with value [property_value].

Explanation: The internal process that is responsible for Kafka communication is using the specified Kafka property value. This informational message is repeated for each Kafka property.

User Response: No response is required.

The process [process_name] is replacing the existing Kafka property [property_name] and value [property_value] with the new property [new_property_name] and value [new_property_value].

Explanation: The internal process that is responsible for Kafka communication is replacing an incompatible Kafka property value. The property is required to connect to Kafka. Normally, the properties that are used to connect to Kafka are provided in a file, such as a Kafka producer properties file. If a Kafka property value is incompatible, the process replaces it. This informational message identifies the existing property name and value that is being replaced and the new property name and value.

User Response: No response is required.

The process [process_name] is adding the Kafka property [property_name] with the value [property_value] to support the Guaranteed Delivery configuration setting.

Explanation: The internal process that is responsible for Kafka communication is adding a Kafka property value to the properties list that it uses. Normally, the properties that are used to connect to Kafka are provided in a file, such as the Kafka Producer properties file. If this process is configured to use Guaranteed Delivery of messages to Kafka, the process might add Kafka properties to meet the Guaranteed Delivery requirement. This informational message identifies a Kafka property and value that is being added to support Guaranteed Delivery.
User Response: No response is required.

**CDCPUB_15026** The process [process_name] is changing the Kafka property [property_name] with the value [property_value] to the new value [new_property_value] to support Guaranteed Delivery.

Explanation: The internal process that is responsible for Kafka communication is changing the value of a Kafka property to support the Guaranteed Delivery setting in the configuration. Normally, the properties that are used to connect to Kafka are provided in a file, such as the Kafka Producer properties file. If this process has been configured for Guaranteed Delivery of messages to Kafka, the process might adjust the input Kafka properties to meet the Guaranteed Delivery requirement. This informational message identifies a Kafka property value that is being changed to another value to support Guaranteed Delivery.

User Response: No response is required.

**CDCPUB_15027** The process [process_name] is adding the Kafka property [property_name] with the default value of [property_value].

Explanation: The internal process that is responsible for Kafka communication is adding the specified Kafka property with the default value to the properties list that it uses. Normally, the properties that are used to connect to Kafka are provided in a file, such as a Kafka Producer properties file. If a required Kafka property is missing, the process will add it. This message indicates the Kafka property is being added and its default value.

User Response: If the default value is not appropriate, add the configuration parameter with another value to the Producer properties file.

**CDCPUB_15028** An attempt to connect to Kafka was unsuccessful because of a Class Not Found exception. A third-party library is probably missing from the generated Java classpath. Error: [error_description].

Explanation: The internal process that is responsible for Kafka communication tested a connection to Kafka but the test failed because of a Class Not Found exception. This error is not fatal. This informational message is issued only when the process tests connectivity to or availability of the Kafka instance and the test fails.

User Response: Typically, this message indicates that not all of the Kafka client library files have been added to the generated classpath. Review the product documentation to determine how to properly add the Kafka client libraries to the environment. Then ensure that all of the Kafka libraries have been included.

**CDCPUB_15029** A test connection to Kafka was unsuccessful because of a TimeOut exception. The Kafka broker might be down or unreachable, or the Kafka Producer properties file might be incorrect. Error description: [error_description].

Explanation: The internal process that is responsible for Kafka communication tested a connection to Kafka but the test failed because of a TimeOut exception. This error is not fatal. This informational message is issued only when the process tests connectivity to or availability of the Kafka instance. The message might indicate that Kafka is not available or that a time-out property in the Kafka Producer properties file is set too low for the environment.

User Response: No response is required.
CDCPUB_15030  A test connection to Kafka was unsuccessful. The Kafka broker might be down or unreachable, or the Kafka Producer properties file might be incorrect. Error description: [error_Description].

Explanation: The internal process that is responsible for Kafka communication tested a connection to Kafka but the test failed. This error is not fatal. This informational message is issued only when the process tests connectivity to or availability of the Kafka instance and the test fails. The Kafka broker might be down or unreachable, or the Kafka Producer properties file might contain an error.

User Response: Verify that the Kafka properties are correctly specified and that the Kafka instance is reachable in the current environment. If the problem persists, contact Informatica Global Customer Support for assistance.

CDCPUB_15031  The process [process_name] requested partition information from Kafka for the topic [topic_name], but the request failed with Kafka error: [error_description].

Explanation: The internal process that is responsible for Kafka communication encountered the specified error while requesting partition information from Kafka for a topic.

User Response: After you review the error description, determine if the error is related to the availability or configuration of the Kafka target environment or if the topic or its partitions are not available on the Kafka server specified in the Kafka Producer properties file. Also, analyze the accompanying log messages to diagnose the error. If the problem persists, contact Informatica Global Customer Support.

CDCPUB_15032  The CDC Publisher did not update any Kafka topics since the last reset.

Explanation: This informational message reports that the CDC Publisher did not update any Kafka topics during the current run or interval. Either no data was available to process, or the data is not newer than the last record that the CDC Publisher sent to Kafka.

User Response: No response is required.

CDCPUB_15033  The CDC Publisher updated the following Kafka topics:

Explanation: This informational message introduces a list of the updated Kafka topics.

User Response: No response is required.

CDCPUB_15034  The CDC Publisher sent [count] messages to the Kafka topic [topic_name] partition [partition_ID]. The last message was written to the topic partition offset [offset] with the key [Kafka_key_value] and includes the source table [table_name] operation [dml_operation] from the stream sequence [sequence_value].

Explanation: This informational message reports the number of messages that the CDC Publisher sent to the specified Kafka topic partition. This message also includes detailed information about the last message written to the topic, including its offset in the topic partition, key value, and the source table operation it contains.

User Response: No response is required.

CDCPUB_15035  An event of type [event_type] for the table [table_name] at the stream sequence [event_sequence_value] recorded the following detailed latency statistics:

Explanation: This informational message introduces latency information for the event that is associated with the specified table and located at the specified sequence position in the change stream.

User Response: No response is required.
<table>
<thead>
<tr>
<th>Message Code</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>CDCPUB_15036</td>
<td>The process [process_name] performed [action] at UNIX Epoch time [time] milliseconds. Duration: [duration_of_action] milliseconds. Additional information: [information]. Explanation: This informational message reports the following information about a process that affected the Change Data Capture event that is identified in message CDCPUB-15035. • The action the process was performing on the event. • The UNIX Epoch time at which the process performed the action on the event. Epoch time is the number of milliseconds between the current time and midnight January 1, 1970 UTC. • The duration, in milliseconds, that has elapsed since the prior action. • Additional information about the process or environment at the time of the action. This information is typically used to determine which processes require the most time to perform actions on an event. User Response: No response is required.</td>
</tr>
<tr>
<td>CDCPUB_15037</td>
<td>The total latency for this Event was [number_ofMilliseconds] milliseconds. Explanation: This informational message provides the total time, in milliseconds, that the CDC Publisher took to process the Change Data Capture event identified in message CDCPUB-15035. User Response: No response is required.</td>
</tr>
<tr>
<td>CDCPUB_15038</td>
<td>The Connector [process_name] will start sending messages to a target entity from the checkpoint [checkpoint_value]. Explanation: The Connector internal helper process that is responsible for sending messages to an external target entity keeps a record of its progress, called a checkpoint. This informational message reports the initial checkpoint value from which the specified Connector process will start sending messages. User Response: No response is required.</td>
</tr>
<tr>
<td>CDCPUB_15039</td>
<td>The Connector process [process_name] with the thread ID [thread_ID] has processed [number_of_operations] operations. Explanation: This informational message reports the number of change operations that the Connector process received. The message can be issued periodically to indicate the streaming activity. User Response: No response is required. You can configure the frequency at which this message is issued. If you need assistance, contact Informatica Global Customer Support.</td>
</tr>
<tr>
<td>CDCPUB_15040</td>
<td>The CDC Publisher wrote [count] messages to file [file_name]. Explanation: This informational message reports the number of messages that the CDC Publisher wrote to the specified output file since the last RESET operation. User Response: No response is required.</td>
</tr>
<tr>
<td>CDCPUB_15041</td>
<td>The process [process_name] determined Kafka serialization settings are expecting message objects of type [object_type]. Explanation: The process determined that the Kafka target is expecting message objects of the specified type. Kafka can use different serializers to support message objects of different types, including strings and byte arrays. The internal process that is responsible for Kafka communication determined the specified type is appropriate for the current environment.</td>
</tr>
</tbody>
</table>
User Response: No response is required. If you do not want to use the specified message object type, check the value.serializer setting in your Kafka producer.properties file, which is used to make this determination. Alternatively, configure the process to not automatically detect the message object type based on the producer.properties file.

**CDCPUB_15042** The process [process_name] will create and send message objects of type [object_type] to Kafka.

Explanation: The internal process that is responsible for Kafka communication is configured to deliver message objects of the specified type to Kafka.

User Response: No response is required. If you do not want to use the specified message object type, edit the appropriate configuration property to use a different Kafka object type.

**CDCPUB_15043** The process [process_name] has loaded [count] topic definitions from the configuration for this environment.

Explanation: The internal process that is responsible for communication with the target has been configured to use the specified number of Kafka topics.

User Response: No response is required. If the number of topics is different from what you expected, check the topic configuration options to verify that all topic definitions have been configured.

**CDCPUB_16001** The command [command_name] was unsuccessful. Error description: [error_description].

Explanation: This error message is typically issued by simple command-line tools to indicate that a command failed to run. The message includes a description of the error.

User Response: Review the system log that the command-line tool produces for more messages related to the problem.

**CPPSVC_00100 to CPPSVC_00103**

**CPPSVC_00100** No localized message factory provided. Message code [message_code]. Parms: [parameters].

Explanation: An internal logic error occurred during CDC Publisher processing. A C++ service attempted to generate a localized message that is associated with the specified message code and parameters, but no localized message factory exists. This message is generated in place of the localized message.

User Response: Contact Informatica Global Customer Support.

**CPPSVC_00101** The localized message with code [message_code] was not found. Parms: [parameters].

Explanation: An internal logic error occurred during CDC Publisher processing. A C++ service attempted to generate a localized message but did not find a localized message that is associated with the specified message code and parameters. This message is generated in place of the localized message.

User Response: Contact Informatica Global Customer Support.
CPPSVC_00102  The process [process_name] with thread ID [thread_ID] encountered error code [error_code] with description [description].

Explanation: During CDC Publisher processing, a C++ service encountered an internal error related to a Java exception. This message provides the process name, associated thread number, and error information returned by Java. The error code is always 0.

User Response: Contact Informatica Global Customer Support.

CPPSVC_00103  Stack trace [stack].

Explanation: During CDC Publisher processing, the stack trace associated with a C++ service encountered an unexpected error.

User Response: Contact Informatica Global Customer Support.

CPPSVC_00104  The request to restart from the end of the PowerExchange CDC Stream identified the following position as the current end of the log: [position]

Explanation: This informational message identifies the restart position that has been calculated when a restart from the end of the PowerExchange change data capture stream has been requested.

User Response: If you do not want to restart from this position, you can reconfigure the environment to restart from a specific location. If necessary, contact Informatica Global Customer Support.

LOGCFG_10001 to LOGCFG_10006

LOGCFG_10001  The Informatica logging component could not create the GMD appender.

Explanation: The Informatica logging component failed to create the appender that is responsible for guaranteed message delivery.

User Response: Analyze the accompanying log messages to diagnose the error and then try to correct the error. If the problem persists, contact Informatica Global Customer Support.

LOGCFG_10002  The Informatica custom Log4j configuration does not include the required appender [appender_name].

Explanation: The Informatica logging component could not find the specified appender defined in the Log4j configuration file. This appender is required.

User Response: Add the missing configuration statements for the appender to the Log4j configuration file. If the problem persists, contact Informatica Global Customer Support.

LOGCFG_10003  The Informatica logging component [component_name] references the appender [appender_name] but this appender is not in the Log4j configuration file. The reference is ignored.

Explanation: The Informatica logging component could not find the specified appender defined in the Log4j configuration file.

User Response: In the Log4j configuration file, add or correct the configuration statements for the missing appender or remove the AppenderRef reference for the appender from the Logger specification. If the problem persists, contact Informatica Global Customer Support.
LOGCFG_10004  The Informatica Logger name [logger_name] in the logging configuration conflicts with the reserved name [reserved_name]. The Logger specification is ignored.

Explanation: The Informatica logging component determined that the Logger name that is specified in the logging configuration file matches a reserved name. Because the Logger cannot have a reserved name, the Logger specification is ignored.

User Response: In the logging configuration file, change the Logger name to a name that is not reserved. If the problem persists, contact Informatica Global Customer Support.

LOGCFG_10005  The Log4j configuration file [file_name] could not be read.

Explanation: The Informatica logging component could not read the specified Log4j configuration file.

User Response: Verify that the file exists and is readable. If the problem persists, contact Informatica Global Customer Support.

LOGCFG_10006  Could not create the Log4j context. Result was null. Does a configuration error exist?

Explanation: The Informatica logging component failed to create the Log4j context object.

User Response: Verify that the Log4j configuration properties are valid. If the problem persists, contact Informatica Global Customer Support.

PWXLIC_00110 to PWXLIC-00999

PWXLIC_00110  Key error. License key [key_value] has a check digit error. Expected check digit [expected_check_digit] but got calculated check digit [calculated_check_digit].

Explanation: The license key is not valid. The check digit is incorrect.

User Response: Enter a valid license key.

PWXLIC_00111  Key error. License key length must be 44 or 64 characters. Key [key_value] has length [number_of_characters].

Explanation: The license key length is not valid. The length must be either 44 or 64 characters.

User Response: Specify a license key value that has a valid length.

PWXLIC_00112  Key error. Hyphen characters in license key [key_value] are not in expected locations.

Explanation: The license key is not valid because it contains hyphens in locations that are not valid within the hexadecimal key values.

User Response: Specify a valid license key that includes hyphens in locations that are valid within hexadecimal key values.

PWXLIC_00113  PowerExchange licensing check found I/O error related to file [key_file_name] operation [operation]. Error description [error_description].

Explanation: An I/O error occurred while PowerExchange was trying to perform a licensing check on the specified license key file. The message reports the file operation in progress when the error occurred and a description of the error. Operation types include open, read, and close.

User Response: Verify that the file name is correct.
<table>
<thead>
<tr>
<th>Code</th>
<th>Description</th>
<th>Explanation</th>
<th>User Response</th>
</tr>
</thead>
<tbody>
<tr>
<td>PWXLIC_00115</td>
<td>Key error. License key [key_value] expired on date [expiration_date]. Current date is [current_date].</td>
<td>The license key expired on the specified date.</td>
<td>Specify a new license key.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>PWXLIC_00116</td>
<td>Key error. License key [key_value] has [number_of_days] days before expiration.</td>
<td>The license key will expire in the specified number of days.</td>
<td>Specify a new license key before the current key expires.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>PWXLIC_00118</td>
<td>Key error. License key [key_value] does not match the platform [platform_type].</td>
<td>The specified license key was issued for a platform other than the platform on which the product will run.</td>
<td>Get a license key that covers the specified platform.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>PWXLIC_00121</td>
<td>Not licensed for feature [feature].</td>
<td>Your license key does not cover the specified feature.</td>
<td>If you need to use the feature, get a license key that allows its use.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>PWXLIC_00122</td>
<td>Key error. License key [key_value] has a vendor code mismatch. Key vendor code [key_vendor_code] does not match the software vendor code [software_vendor_code].</td>
<td>The license key is not valid because it has been issued for a vendor other than the vendor for which the software has been supplied.</td>
<td>Verify that you are using the correct license key, which covers the vendor of the supplied software. If you need to get a new key, contact Informatica Global Customer Support.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>PWXLIC_00800</td>
<td>Key error. License key [key_value] is not a valid hexadecimal value because it contains the character [character].</td>
<td>The license key is not valid because it contains the specified non-hexadecimal character. The license key must be a hexadecimal value.</td>
<td>Specify a license key that is a valid hexadecimal value. Ensure the key value does not contain the specified non-hexadecimal character.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>PWXLIC_00801</td>
<td>Key error. The 64-character license key [key_value] contains an indicator that identifies the key as invalid.</td>
<td>The 64-character license key is not valid, as indicated by a bit in the key value.</td>
<td>Specify a valid long license key.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>PWXLIC_00802</td>
<td>License environment variable [variable_name] is missing or has no value.</td>
<td>The environment variable that points to the PowerExchange license key file is not present or does not contain a value.</td>
<td>Specify the correct path to the license key file in the environment variable.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>PWXLIC_00999</td>
<td>PowerExchange license checking encountered the internal logic error [error_description].</td>
<td>An internal logic error occurred during PowerExchange license checking. This message includes a description of the error.</td>
<td></td>
</tr>
</tbody>
</table>
User Response: Contact Informatica Global Customer Support.

PWXRDR-01000 to PWXRDR-01301

**PWXRDR_01000**  The PWX Reader has initialized. Version [version].
Explanation: The PWX Reader Service has initialized. This message includes the PWX Reader version.
User Response: No response is required.

**PWXRDR_01001**  The PWX Reader is terminating.
Explanation: The PWX Reader service is terminating. This message does not indicate an error condition.
User Response: No response is required.

**PWXRDR_01002**  Required PWX Reader configuration key [key_name] is missing.
Explanation: The specified PWX Reader configuration key name is required but is missing.
User Response: Enter the configuration key name.

**PWXRDR_01003**  PWX Reader configuration key [key_name] has an invalid value [value].
Explanation: An invalid value is specified for the specified PWX Reader configuration key.
User Response: Correct the configuration key value.

**PWXRDR_01004**  PWX Reader configuration key [key_name] with value [value] conflicts with configuration key [key_name] with value [value].
Explanation: The PWX Reader configuration keys that are identified in this message have conflicting values.
User Response: Correct one or both of the configuration key values so that they are compatible.

**PWXRDR_01005**  PWX Reader detected that no tables are selected for change data extraction.
Explanation: The PWX Reader configuration does not match any tables from which PowerExchange captures changes. This situation might be caused by user-specified filtering criteria or an invalid extraction map schema name.
User Response: Correct the PWX Reader configuration so that it matches at least one table from which PowerExchange captures changes.

**PWXRDR_01006**  PWX Reader termination has waited [seconds] for the following methods to complete: [methods].
Explanation: The PWX Reader has waited the specified number of seconds to terminate. It cannot terminate until the specified methods complete processing.
User Response: No response is required.
PWXRDR_01007  PWX Reader configuration key [old_key_name] is deprecated. Use the new configuration key [new_key_name].  
Explanation: The specified PWX Reader configuration key has been deprecated. Use the new configuration key that this message supplies.  
User Response: Enter a value for the new key in the configuration.

PWXRDR_01100  Connecting to PowerExchange location [location] to collect metadata.  
Explanation: The PWX Reader service is connecting to the specified PowerExchange location to collect metadata about the source objects for which to extract change data.  
User Response: No response is required.

PWXRDR_01101  Connecting to the PowerExchange location [location] for change data extraction.  
Explanation: The PWX Reader service is connecting to the specified PowerExchange location to extract change data.  
User Response: No response required.

PWXRDR_01102  Connection is using the CAPI_CONNECTION [capi_connection_name].  
Explanation: The PWX Reader service is configured to use the specified CAPI_CONNECTION statement to extract change data from PowerExchange. The CAPI_CONNECTION statement is defined in the PowerExchange DBMOVER configuration file on the PowerExchange Listener system.  
User Response: No response is required.

PWXRDR_01104  Connection is using the PowerExchange extraction map schema name [schema_name].  
Explanation: The PWX Reader service is configured to use the specified schema name portion of extraction map name to identify the extraction maps to use for reading change data from source tables. PowerExchange generates this schema name when creating extraction maps.  
User Response: No response is required.

PWXRDR_01105  Connecting to PowerExchange location [location] to get current restart position.  
Explanation: The PWX Reader service is connecting to the specified PowerExchange location to determine the current restart position for extraction processing.  
User Response: No response is required.

PWXRDR_01106  Restart position [hex_restart_position].  
Explanation: This message reports the restart position of PWX Reader extraction processing in the PowerExchange change stream. The restart position is expressed as a hexadecimal restart token.  
User Response: No response is required.

PWXRDR_01107  Connection is using the PowerExchange SCLI connect string [connect_string].  
Explanation: The PWX Reader service is using the specified SCLI connect string to connect to PowerExchange.  
User Response: No response is required.
Explanations:

- **PWXRDR_01110**: Extractions maps selected for [count] source tables.
  - **Explanation**: After collecting metadata about the source tables, the PWX Reader identified the extraction maps to use for extracting change data for the specified number of source tables. The extraction map names are listed in subsequent PWXRDR-1111 messages.
  - **User Response**: No response is required.

- **PWXRDR_01111**: Extraction map [extraction_map_name].
  - **Explanation**: The PWX Reader service of the PowerExchange CDC Publisher is extracting change data for the source table that is associated with the specified extraction map. The extraction map name has the format `xminstance.mapname_tablename`.
  - **User Response**: No response is required.

- **PWXRDR_01112**: Extraction map [extraction_map_name] primary key columns [column_name1,column_name2,...].
  - **Explanation**: The PWX Reader service is extracting change data for the source table that is associated with the specified extraction map. This message identifies the primary key columns for the table.
  - **User Response**: No response is required.

- **PWXRDR_01113**: Not using extraction map [extraction_map_name] for change data extraction because it contains no qualifying columns.
  - **Explanation**: The PWX Reader service will not extract change data based on the specified extraction map because the extraction map contains no columns that qualify for extraction processing.
  - **User Response**: No response is required.

- **PWXRDR_01114**: Not using extraction map [extraction_map_name] for change data extraction because column filtering has excluded all columns in the map.
  - **Explanation**: The PWX Reader service will not extract change data based on the specified extraction map because column filtering has excluded all columns in the map from extraction processing.
  - **User Response**: No response is required.

- **PWXRDR_01115**: Primary key column [column_name] in extraction map [extraction_map_name] is not being extracted because of filtering criteria.
  - **Explanation**: Data is not being extracted from the specified primary key column based on your configuration.
  - **User Response**: No response is required.

- **PWXRDR_01200**: Beginning of PowerExchange informational messages.
  - **Explanation**: This message introduces a set of informational messages that PowerExchange is returning to the PWX Reader.
  - **User Response**: No response is required.

- **PWXRDR_01201**: [info_message_text]
  - **Explanation**: The PWX Reader received the specified informational message from PowerExchange.
  - **User Response**: No response is required.
<table>
<thead>
<tr>
<th>Code</th>
<th>Description</th>
<th>Explanation</th>
<th>User Response</th>
</tr>
</thead>
<tbody>
<tr>
<td>PWXRDR_01202</td>
<td>End of PowerExchange informational messages.</td>
<td>This message indicates the end of a set of informational messages that PowerExchange returned to the PWX Reader service.</td>
<td>No response is required.</td>
</tr>
<tr>
<td>PWXRDR_01300</td>
<td>Command [command] received.</td>
<td>The PWX Reader service received the specified command.</td>
<td>No response is required.</td>
</tr>
<tr>
<td>PWXRDR_01301</td>
<td>Command [command] not recognized.</td>
<td>The PWX Reader service did not recognize the specified command as a valid command.</td>
<td>No response is required.</td>
</tr>
</tbody>
</table>
INDEX

A
administering PowerExchange CDC Publisher
  overview of administrative tasks 55
Apache Kafka targets
  CDC Publisher 10
  data-streaming considerations 35
  overview 34
  preparing the target environment 34
Avro schemas
  reporting generated schemas for source tables 56

C
CDC Publisher
  architecture of Java-based component 10
command reference
  commands for the command-line utilities 62
  PwxCDCInfo command 64
  PwxCDCPublisher command 63
command-line utilities
  commands reference 62
  PwxCDCInfo command syntax 64
  PwxCDCPublisher command syntax 63
configuration files
  cdcPublisherAvro.cfg 38
  cdcPublisherCommon.cfg 38
  cdcPublisherKafka.cfg 40
  cdcPublisherPowerExchange.cfg 41
  customizing the sample files 37
configuration properties
  properties for Avro schema format and message encoding 38
  properties for extracting data from PowerExchange 41
  properties for Kafka targets 40
  properties for the PowerExchange CDC Publisher host and port 38
Configuring PowerExchange CDC Publisher
  overview of configuration tasks 36

D
data streaming
  before starting PwxCDCPublisher streaming 48
  operational considerations 49

E
encrypting a password 59
environment variables
  configuring 22
extraction maps
  filtering extraction maps and source objects 45

G
guaranteed delivery 27

I
Informatica Intelligent Streaming
  integration considerations 47
installation
  installing on Linux or Windows 20
  overview 18

K
Kafka targets
  preparing the target environment 34

L
Log4j 2 message logging
  configuring for the PowerExchange CDC Publisher 46

M
message logging
  configuring Log4j 2 logging 46
  messages for data delivery
    message content and format 25
  monitoring PowerExchange CDC Publisher
    overview 51

P
password encryption
  using the PwxCDCAdmin utility 59
PowerExchange CDC Publisher utilities
  overview 16
PowerExchange environment
  alternative configurations with CDC Publisher 11
  configuration tasks 31
  configuring connectivity to PowerExchange 32
PowerExchange CDC Admin utility
  adding a source table 58
  clearing Avro schemas 56
  encrypting a password 59
  performing administrative tasks 55
  regenerating Avro schemas 56
  reporting Avro schemas for source tables 56
  shutting down the CDC Publisher 59
PwxCDCInfo utility
- generating statistics for topics and processes 52
- reporting the status of CDC Publisher processes 51

PwxCDCPublisher utility
- operational considerations 49
- prerequisites to starting 48
- restarting change stream processing 60

PwxCDCPublisher utility
- starting a CDC Publisher process 48

R
- restart
  - apply processing restart 28
  - CDC Publisher restart in change stream 28
  - restarting the PwxCDCPublisher utility in the input change stream 60

S
- shutting down the CDC Publisher
  - with the PwxCDCAdmin utility 59
- source tables
  - adding a table to a change stream 58

source tables (continued)
- handling changes to source table definitions 56

start a CDC Publisher process
- with the PwxCDCPublisher utility 48

statistics
- generating with the PwxCDCInfo utility 52
- status reporting
  - for CDC Publisher processes 51

U
utilities
- commands reference 62
- overview of PowerExchange CDC Publisher utilities 16
- PwxCDCInfo command syntax 64
- PwxCDCPublisher command syntax 63