ATTUNITY REPLICATE

Setup and User Guide

Version 6.0

November 05, 2017

Attunity Replicate Setup and User Guide, Version 6.0

Primary Author: Charlton Book

Contributors: Allen Reiter, Tzachi Nissim, Dror Harari, Ofer Haramati, Sami Zeitoun, Boris Goberman, Gilad Sherry, Hanna Zaboura

Contents

5 | Using the Attunity Replicate Console

6 | Getting Started: An Attunity Replicate Tutorial

7 | Designing Tasks

12 | Working with Tasks at Runtime

13 | Monitoring and Controlling Replication Tasks

1 | Introduction

This section describes the main concepts of data replication and the major components of Attunity Replicate.

Note The term "endpoint" is used generically throughout this guide to refer to a data repository that can be used as a source and/or target in an Attunity Replicate task. Examples of such repositories include relational databases (such as Oracle), NoSQL databases (such as MongoDB) and files.

In this chapter:

Replication Explained

Replication is a process that keeps two or more collections of computerized information identically synchronized. It facilitates:

- **Load reduction:** Keeping a complete or partial copy of a collection on a different server reduces the load on the main server.
- **Improved service:** Accessing a copy of the data can provide better service to users than having them access the original data..
- **Restricted data access:** If some users should only have access to a subset of data, replicating only part of a collection makes it easy to enforce security restrictions.
- **Geographic distribution:** Making only a subset of data relevant to a specific node (or location) available is beneficial in widely distributed enterprises (such as a chain of retail stores or warehouses). You can still make all data available at a central location for less frequent use.
- **Disaster Recovery:** Keeping a copy of the main data available allows for setting up rapid fail-over clusters (the capability to switch over to a redundant or standby computer server in case the main system fails).

- **"Cloud" computing:** Replicating data allows for implementing what is commonly known as cloud computing (the on-demand storage, management, and processing of Internet-based data).

The information replicated is stored as files or in a database. In the case of files, the structure and content of a file are known only to the specialized programs that use the file. Databases are managed by database management systems (DBMS) that make use of standardized descriptions of the structure of the information (such as tables, columns, rows, and data types). These descriptions are known collectively as metadata and allow a general-purpose replicator to carry out relevant operations (for example filtering and data transformations) without the need to know anything about the contents or "meaning" of the data. Because file systems do not contain metadata, operations available for replication are more limited.

During replication, a collection of data is copied from system A to system B, where A is known as the source (for this collection) and B is known as the target. A system can be a source, a target, or both (with certain restrictions). A complex replication topology has a number of sources, targets, and data collections defined.

The replication process must account for the fact that source data may be changing while being copied. It is not possible to make or maintain copies instantaneously and to stop the source computer to "freeze" the information. Therefore, replication must account for:

- **Integrity:** The target data must reflect the complete result of all changes made to the source data during the replication process.
- **Consistency:** If a change affects different tables, rows, or files, the copy must reflect these changes consistently across all affected tables, rows, or files.
- **Latency:** The replication process must aim at keeping latency at a minimum. Ideally, it should not exceed a few seconds.

Attunity Replicate

Attunity Replicate is a simple, powerful, easy-to-implement solution that provides replication between various endpoints. Replicate lets you:

- Load data efficiently and quickly to operational data stores/warehouses.
- Create copies of production endpoints.
- Distribute data across endpoints.

Replicate has high throughput, speed, and scale. It is designed to scale and support large scale enterprise data replication scenarios with a multi-server, multi-task, and multi-threaded architecture.

Replicate consists of a Web-based console and a replication server to replicate data across heterogeneous data sources. It provide users with instant visibility into current and historical exceptions, status, performance, and resource usage information.

Replicate can execute replication tasks between enterprise endpoints including Oracle, Microsoft SQL Server, and IBM DB2. It uses a "Click-2-Replicate" design that simplifies the replication process by automating the steps required to build a replication solution.

When you run a task in Replicate, you can select between:

- Full Load Replication: Creates files or tables at the target endpoint, automatically defines the metadata that is required at the target, and populates the tables with data from the source
- Change Processing, also called Change Data Capture (CDC): Captures changes in the source data or metadata as they occur and applies them to the target endpoint as soon as possible in near-real time
- Replication is log based, which means that it reads only the changes. This reduces the impact on the source endpoints.

Limitations

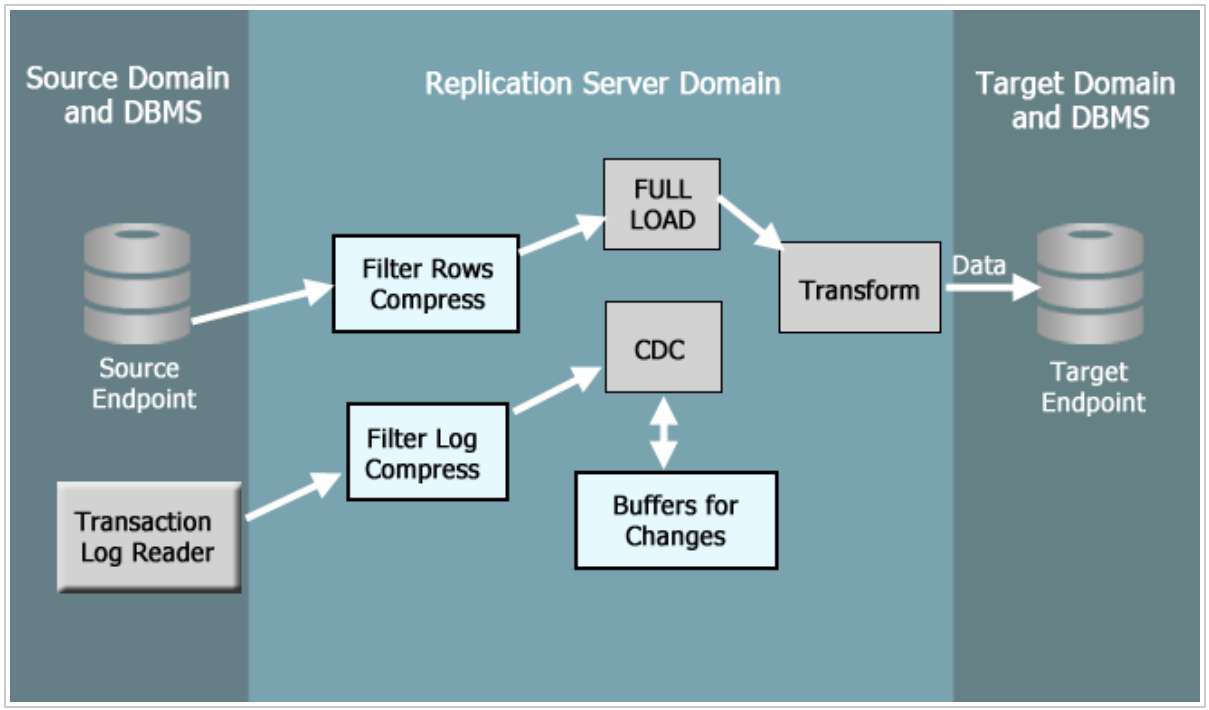
The following limitations apply:

- Attunity Replicate performs data transfer between source and target endpoints using Unicode, which supports characters that need a maximum of three bytes. Attunity Replicate does not support characters requiring four bytes, such as mathematical symbols and emojis.
- Replicate does not support replication of Primary Keys that are LOB data types.
- When the Limit LOB size to option is enabled, replication of structured data LOBs (e.g. XML, JSON, IMAGE, etc.) may truncate (and thereby invalidate) the structured data in the target LOB.
- When Replicate creates a new table in the target endpoint, it defines only *one* index on the table. The index will either be the Primary Key or the first unique key (according to alphabetical order) of the table. No other indexes will be defined in the target. If additional indexes are required, these will need to be created manually.
- LOB columns are always created as nullable on the target database. If you create the target table(s) manually, then you must set all LOB columns to nullable.
- The Preserve transactional integrity option is supported with the Oracle target endpoint only.
- If you stop a task after Full Load completes, perform some changes on the source tables, and later resume the task from timestamp (by selecting the **Start processing changes from** run option), some changes may not be replicated to the target. This usually only happens if the transaction logs in the source database have been deleted due to a log purge policy. In this case, Replicate will resume the task from the last change in the current transaction log.

System Architecture

The following diagram shows the basic architecture of Attunity Replicate.

Figure 1.1 | Attunity Replicate System Architecture



In this diagram, the source data and metadata are part of the source server. The transaction log reader can be on the source server (for efficiency) or on the Attunity Replicate server (for zero footprint on the source). Filtering and compression of the source rows/logs can occur on the source or Attunity Replicate servers.

In the initial load process, Attunity Replicate reads a filtered stream of rows (with relevant columns only) and passes them to the transformation process for further filtering and subsequent writing to the target endpoint (in the expected output format).

The CDC process obtains a stream of filtered events or changes in data or metadata from the transaction log file. It then buffers all changes for a given transaction into a single unit before forwarding them to the target when the transaction commits. During the initial load process, CDC also buffers all changes that occur within a transaction until all affected tables have been loaded.

The Designer/Console server, which is part of the Replication server, is a Web-based application that serves as the user interface for dealing with designing or modifying the replication system and displaying and controlling its operation.

Replication Tasks

Each instance of a table synchronization activity comprises a task in Attunity Replicate. You define a task using the browser-based Attunity Replicate Console. When defining a task, you specify:

- The source and target endpoints
- The source and target tables to be kept in sync
- The relevant source table columns
- The filtering conditions (if any) for each source table as Boolean predicates (in SQLite syntax) on the values one or more source columns
- The target table columns (optionally), including their data types and values (as expressions or functions over the values of one or more source or target columns, using SQL syntax). If not specified, Replicate uses the same column names and values as the source tables, with default mapping of the source DBMS data types onto the target DBMS data types. Replicate automatically takes care of the required filtering, transformations, and computations during the Load or CDC execution.

When a task is defined, you can activate it immediately. Replicate automatically creates and loads the target tables with the necessary metadata definitions and activates the CDC. Using the Attunity Replicate Console, you can then monitor, stop, or restart the replication process.

Using Multiple Tasks

You can define and activate several replication tasks at once. This is best if the tasks:

- Have different source tables
- Share some source tables but have different filtering conditions on the source rows
- Update different target tables

Updating the same target table and row by two different replication tasks would not be good practice and may cause unpredictable results.

The different replication tasks work independently and run concurrently. Each has its own Initial Load, CDC, and Log Reader processes.

Full Load and CDC Processes

The full load process creates files or tables at the target endpoint, automatically defines the metadata that is required at the target, and populates the tables with data from the source. Unlike the CDC process, the full load process loads the data one entire table or file at a time, for maximum efficiency.

The source tables may be subject to update activity during the Load process. *However, there is no need to stop processing in the source*. Replicate automatically starts the CDC process as soon as the load process starts. It does not apply the changes to the target until after the load of a table completes because the data on the target might not be consistent while the load process is active. At the conclusion of the load process, however, Replicate guarantees consistency and integrity of the target data.

If the load process is interrupted, it continues from wherever it stopped when restarted.

You can add new tables to an existing target without reloading the existing tables. Similarly, you can add or drop columns in previously populated target tables without reloading.

The CDC process captures changes in the source data or metadata as they occur and applies them to the target endpoint as soon as possible in near real time. It captures and applies the changes as units of single committed transactions and can update several different target tables as the result of a single source commit. This guarantees transactional integrity in the target endpoint. The CDC process for any file or table starts as soon as the data load process for the file or table begins.

CDC operates by reading the recovery log file of the source endpoint management system and grouping together the entries for each transaction. The process employs techniques that ensure efficiency without seriously impacting the latency of the target data. If the CDC process cannot apply the changes to the target within a reasonable amount of time (for example when the target is not accessible), it buffers the changes on the Replication server for as long as necessary. There is no need to re-read the source DBMS logs, which may take a long time.

Replication Topologies

Attunity Replicate supports the following topologies for replication tasks:

- One to One
- Logical Independence
- Hub and Spoke

One to One

In a one-one topology, there is one source and one target endpoint. When the source and target endpoints are distinct, Attunity Replicate guarantees transactional integrity and consistency. If you use two different replication tasks, the endpoints may switch roles, allowing two-way synchronization.

Caution: If the same row in a table is updated by two different replication tasks, the result of two-way synchronization may be unpredictable. A problem can occur even if two different rows are referentially related, that is if some application updates a row based on reading a value in a different row. If the rows are updated concurrently on the source and the target, the result may be unpredictable[1] . Such occurrences are rare, but they can occur.

[1]CDC has no way of knowing exactly when a row was read by an application on one system relative to its having been changed on another system. Read operations are typically not logged.

Logical Independence

Two-way replication works best when updates of a row on a source and on a target are entirely autonomous and do not affect each other. There is an assumption that any table or a horizontal or vertical segment of a partitioned table can only be updated in one source. Attunity Replicate allows updating the same row in several places, but in this case, the columns being updated must be distinct. Another assumption is that if a data value in one row depends on or is derived from a value in another row, the values can be changed only on the same server but nowhere else (except by the Replicator). This is called logical independence. With logical independence, concurrent update conflicts cannot occur during replication.

Hub and Spoke

Many-to-one and one-to-many relationships can be combined into a hub-and-spoke topology, which allows the merging of data into multiple targets and then distributing to other targets. It does not allow cycles or multiple paths for propagating changes. The hub-and-spoke topology is that of an acyclic directed graph.

2 | Security Considerations

Attunity Replicate is tasked with replicating data within an organization, a task which involves reading from source endpoints (such as databases) and writing to target endpoints (including databases, files, and queuing systems).

This section provides important information for protecting the data that Attunity Replicate stores and replicates.

In this chapter:

Securing Access to the Attunity Replicate Web UI

Attunity Replicate offers the following Web UI configurations:

- A Windows-based Attunity Replicate UI Server service which offers granular user authorization based on a user's Active Directory identity and group membership. This service provides user interface functionality and communicates with the backend Attunity Replicate Server (on Windows or Linux).

  Connecting to this server is done using HTTPS with Windows Authentication.
- The Attunity Replicate Server on Windows or Linux can also serve the Web UI directly, but supports just a single user with a fixed role of administrator ('admin').

  Connecting to this server is done using HTTPS with Basic Authentication.

  See also Configuration 3: Replicate UI Console and Replicate Server Running on Linux.

In line with current industry security standards, the Attunity Replicate web user interface enforces the use of HTTPS to protect against eavesdropping and data leakage. Using HTTPS requires a valid server certificate to be installed on the Attunity Replicate server machine.

Setting Up Replicate Console HTTPS Support

Industry-standard security practices dictate that web user interface for enterprise products must use secure HTTP (HTTPS). Attunity Replicate enforces the use of HTTPS and will not work if HTTPS is configured incorrectly.

As Attunity Replicate uses the built-in HTTPS support in Windows, it relies on the proper setup of the Windows machine it runs on to offer HTTPS access. In most organizations, the IT security group is responsible for generating and installing the SSL server certificates required to offer HTTPS. It is strongly recommended that the machine on which Replicate is installed already has a valid SSL server certificate installed and bound to the default HTTPS port (443).

Checking if an SSL Certificate is Installed

To check whether an SSL certificate is installed, you can use the following command:

```
netsh http show sslcert | findstr /c:":443 "
```

If an SSL certificate is installed, the output should look like this:

```
netsh http show sslcert | findstr /c:":443 "
IP:port : 192.168.1.13:443
IP:port : 192.168.1.11:443
IP:port : [fe80::285d:599c:4a55:1092%11]:443
IP:port : [fe80::3d0e:fb1c:f6c3:bc52%23]:443
```

With a valid SSL certificate installed, the Attunity Replicate web user interface will automatically be available for secure access from a web browser using the following URL:

`https://<machine-name>/`AttunityReplicate

Using the Self-Signed Certificate

Due to the way the HTTPS protocol works, there is no way for Attunity Replicate to automatically provide and install a valid SSL server certificate. Still, in the event that no SSL server certificate is installed, Attunity Replicate automatically generates and installs a self-signed SSL server certificate (as a temporary measure). This certificate is generated on the Replicate machine and cannot be exported or used elsewhere.

It should be noted that browsers do not consider the certificate to be valid because it was not signed by a trusted certificate authority (CA).

When connecting with a browser to a server that uses a self-signed certificate, a warning page is shown informing you that the connection is not secure or similar (depending on the browser).

The warning page informs you that the certificate was signed by an unknown certificate authority. All browsers display a similar page when presented with a self-signed certificate. If you know that the self-signed certificate is from a trusted organization, then you can instruct the browser to trust the certificate and allow the connection. Instructions

on how to trust the certificate vary between browsers and even between different versions of the same browser. If necessary, refer to the help for your specific browser.

Note Some corporate security policies prohibit the use of self-signed certificates. In such cases, it is incumbent upon the IT Security department to provide and install the appropriate SSL server certificate (as is the practice with other Windows products such as IIS and SharePoint). If a self-signed certificate was installed and needs to be removed, then the following command can be used:

```
$ netsh http delete sslcert ipport=192.168.1.13:443
```

where `ipport` should be replaced with the `ip:port` combination generated by the `netsh` command shown in Checking if an SSL Certificate is Installed.

Setting Up Attunity Replicate Server HTTPS Support

The Attunity Replicate Server which runs on both Windows and Linux uses the OpenSSL HTTPS implementation. The Attunity Replicate Server automatically generates a self-signed certificate server but it allows you to replace it with a server certificate signed by a trusted certificate authority.

Replacing the Self-Signed SSL Certificates on Linux

When Attunity Replicate Server starts for the first time, it checks the ***<product-dir>*/ssl/data** directory for the presence of certificates. If there are no certificates, it will create the following self-signed certificates:

- **agent-ca.pem** - The CA certificate
- **agent-certificate.pem** - The public certificate
- **agent-private-key.pem** - The private key data
- **agent-private-key-passphrase.dat** - The private key passphrase

You can replace these SSL certificates with you own certificates as follows:

1. Stop the Attunity Replicate Server service.
2. Create the required certificates using names that are identical to the certificates listed above.
3. Copy the certificates to the **ssl** directory (***<product-dir>*/ssl/data** by default).
4. Edit the **agent-private-key-passphrase.dat** file as follows:

   `/clear:`*`PRIVATE_KEY_PASSWORD`*

   Example:

   `/clear:12345`

   When Attunity Replicate Server starts it will scramble the private key passphrase as shown in Examples of the Scrambled Private Key Password.
5. Start the Attunity Replicate Server service.

For information on stopping and starting Attunity Replicate Server, see Attunity Replicate on Windows: Installing, Upgrading and Uninstalling and Attunity Replicate on Linux: Installing, Upgrading and Uninstalling.

Examples of the Scrambled Private Key Password

The scrambled private key passphrase stored in the **agent-private-key-passphrase.dat** file will look similar to this:

```
{S:DEA326D0DF190430975DE44CFBD6FDFD21883C10E7651081B3B5A0A7404BB97DB520876F
60390B51300C831C82DE871CF8BA22393D8DD9B359DD5A93C5956710AD2546E188155482452
235C5D91B430D151E3DDA7381CA3E}
```

Replacing the Self-Signed Certificate on Windows

The instructions below are intended for organizations who wish to replace the self-signed certificate generated by the Replicate UI Server on Windows with their own certificate. This is achieved by removing the self-signed certificate and then importing the new certificate.

See also Setting Up Replicate Console HTTPS Support.

Before starting, make sure that the following prerequisites have been met:

- The replacement certificate must be a correctly configured SSL PFX file containing both the private key and the certificate.
- The common name field in the certificate must match the name browsers will use to access the machine.

To remove the self-signed certificate created by Attunity Replicate:

1. Stop the Attunity Replicate Server and Attunity Replicate UI Server services.
2. Open a command prompt (using the "Run as administrator" option) and change the path to the Replicate **bin** directory. The default path is C:\Program Files\Attunity\Replicate\bin.
3. Run the following command:
4. RepUiCtl.exe certificate clean

To import your own certificate:

1. Run mmc.exe to open the Microsoft Management Console.
2. From the **File** menu, select **Add/Remove Snap-in**.

   The **Add or Remove Snap-ins** dialog box opens.
3. In the left pane, double-click **Certificates**.

   The **Certificates snap-in** wizard opens.
4. Select **Computer account** and then click **Next**.
5. In the **Select Computer** screen, make sure that **Local computer** is selected and then click **Finish**.

6. Click **OK** to close the **Add or Remove Snap-ins** dialog box.
7. In the left pane, expand the **Certificates** folder. Then, right-click the **Personal** folder and select **All Tasks>Import**.
8. In the **File to Import** screen, select your PFX certificate file. Note that by default the **Open** dialog box displays CER files. In order to see your PFX files, you need to select **Personal Information Exchange** from the drop-down list in the bottom right of the dialog box.
9. Click **Next** and enter the private key password.
10. Continue clicking **Next** until you reach the **Completing the Certificate Import Wizard** screen. Then click **Finish** to exit the wizard.
11. In the **Personal> Certificates** folder, double-click the newly imported certificate.

    The **Certificate** dialog box opens.
12. Scroll down the **Details** tab until you see the **Thumbprint** details and copy them to the clipboard.
13. Open a command prompt and run the following commands:

    Syntax:

    ```
    ¢ netsh http add sslcert ipport=0.0.0.0:443 certhash=[YOUR_CERTIFICATE_
    THUMBPRINT_WITHOUT_SPACES] appid={4dc3e181-e14b-4a21-b022-59fc669b0914}
    ```

    Example:

    ```
    netsh http add sslcert ipport=0.0.0.0:443
    certhash=5f6eccba751a75120cd0117389248ef3ca716e61 appid={4dc3e181-e14b-
    4a21-b022-59fc669b0914}
    ```

    Syntax:

    ```
    ¢ netsh http add sslcert ipport=[::]:443 certhash=[YOUR_CERTIFICATE_
    THUMBPRINT_WITHOUT_SPACES] appid={4dc3e181-e14b-4a21-b022-59fc669b0914}
    ```

    Example:

    ```
    netsh http add sslcert ipport=[::]:443
    certhash=5f6eccba751a75120cd0117389248ef3ca716e61 appid={4dc3e181-e14b-
    4a21-b022-59fc669b0914}
    ```
14. Close the command prompt and Microsoft Management Console.
15. Start the Attunity Replicate Server and Attunity Replicate UI Server services.

Changing the Server Password

The Attunity Replicate Server has a fixed 'admin' user with an automatically generated random password that is stored in the mk.dat file. Although the password is unknown, it is unique and safe. The Attunity Replicate UI Server services always connects to the Attunity Replicate Server services using the 'admin' user. When both services run on the same machine the admin password is accessible to both servers, so there is no need to specify this password explicitly.

When Attunity Replicate Server runs on a different machine or when a remote Attunity Replicate client needs to communicate with a remote Attunity Replicate Server, the server password must be known to both sides.

The following sections explain how to change the replication server password so that it can be provided remotely.

To change the server password:

1. To set the password using a script, use the following command:
   `repctl SETSERVERPASSWORD <new_password>`
2. To set the password interactively:
   a. Run the following command:
      `repctl`
   b. Press [Enter] and then type the following:
      `SETSERVERPASSWORD <new_password>`
   c. Press [Enter] again to set the password.
3. Restart the Attunity Replicate services (Windows) or the Attunity Replicate daemon (Linux).

Note When the AttunityReplicate .NET UI Server is running on one machine and the AttunityReplicate Server is running on another, the AttunityReplicate Server password must be the same on both machines. The password is used during the SSL handshake to establish a secure connection between the participating machines.

Protecting Replicate Passwords

Replicate stores secrets (e.g. passwords and keys) in its internal repository, enabling it to perform secure operations during runtime, such as connecting to an endpoint, connecting to a web server, connecting to an email server, connecting to a remote file transfer server, and so on.

As a rule, all UI values displayed as asterisks are stored encrypted and never transmitted or exposed by the API. For instance, when exporting task definitions, all passwords are encrypted and can only be decrypted if the source machine's mk.dat file - and possibly also the account or machine identity (on Windows) - is copied to the target machine. See also Master Key Considerations when Exporting and Importing Tasks.

Secrets that appear in Replicate settings are stored in an encrypted form with the following properties:

- The secret is encrypted using the AES-256 encryption algorithm.
- The encryption key, also known as the 'master key', is a 256-bit key that is stored (and protected as described later) in the master key file (**mk.dat**).

- The encryption process uses a salt that depends on the name and type (or 'context') of the protected entity. This prevents reuse attacks.
- The encryption process uses a nonce so that the same secret is always encrypted differently. This prevents dictionary attacks.

When exported to a JSON file, secrets appear in the following format:
"{Zxhhhhhhhhhhhhhhhhhhhhhhhhhhhhhh}"

Where:

- 'Z' is a fixed character
- 'x' a protection method indicator
- 'hhhhhh…' is a hexadecimal representation of the encrypted secret

Upon import, if a secret is provided in clear text (i.e. not in the format shown above), it is automatically encrypted and stored in the protected format.

The master key used in encrypting secrets is stored in the master key file (**mk.dat**) described below.

The Master Key File

The master key file is a binary file that stores the root secrets of Replicate. These root secrets currently include:

- The AES 256-bit master key which is used to encrypt passwords and other secrets in Replicate settings.
- The server admin password which is used (by the UI server) to access the replication server remotely.
- The default location for the master key file is the product data folder (typically "<product-dir>/data" unless overridden with the "-d" command line option). If the server admin password does not exist, it is automatically created with randomly generated passwords that are safe but unknown. The user can change the server admin password as well as the master key to known values if needed (e.g to connect to the replication server remotely).

  For more information, see Changing the Server Password.

  Note If the **mk.dat** file is manually moved to the "<product-dir>/bin" folder, then Replicate will read it from there instead of from the "data" folder.

When Replicate is set to run on a cluster using the same settings and storage, the **mk.dat** file should also be the same (or shared). Similarly, if the Replicate settings are exported in order to be moved to a different machine, the **mk.dat** file should also be moved in order to avoid the need to reenter all secrets.

The procedure for changing the master key as well as measures that can be taken to protect the file containing the master key are described below in Changing and Protecting the Master Key.

Changing and Protecting the Master Key

This section describes how to change the master key as well as how to prevent unauthorized access to the master key file.

Changing the Master Key Replacement

The master key is originally a randomly generated 256-bit key. It can be changed as described below immediately after installation with no consequences. However, if you change the master key *after* endpoints or tasks are already configured, all stored secrets will no longer be valid (as they were encrypted using the old key). Therefore, after changing the master key, you need to reenter the passwords in all the relevant places.

To change the Master Key

1. Stop any running tasks.
2. Stop the Replicate services.
3. Open a command prompt as an administrator.
4. Change the working directory to the product "bin" directory and then issue the following command:

   ```
   repctl setmasterkey <your_new_master_key> [master_key_scope=<scope>]
   ```

   Note Note that if used interactively, this may leave a trace in the shell history file. To prevent this, you can use the command interface:

   ```
   repctl {enter}
   setmasterkey <your_new_master_key> [master_key_scope=<scope>] {enter}
   quit {enter}
   ```

   See Protecting the Master Key File from Misuse for the `master_key_scope` options.

   Example:

   ```
   repctl setmasterkey 78543043vuiyfyrf64454555jy65 master_key_scope=1
   ```
5. Start the Attunity Replicate**Server** service.
6. Re-enter the access passwords in all endpoints.
7. Start the Tasks.

Protecting the Master Key File from Misuse

Access to the **mk.dat** file is restricted to administrators (Windows) or to the users in the group under which the product was installed (Linux). Care should be taken to maintain this file protection.

The **mk.dat** file is always encrypted by some built-in, fixed key. On Windows, there are two additional options for preventing unauthorized access and use of the **mk.dat** file. These are as follows:

- Tying the **mk.dat** to the Machine Profile - With this option, the **mk.dat** file can only be used on the Windows machine where it was created. Using the file from a different

machine will generate an error. This option is not appropriate for clusters that share the **mk.dat** file as only one machine will be able to read it.

- Tying the **mk.dat** to a User Profile - With this option, the **mk.dat** file can only be used under the identity that was used while creating it (typically the user that installed the product). For this option to work, the Replicate services must be modified to run under the same user. If the user has a roaming profile, then the **mk.dat** file can be used on multiple machines (e.g. in a cluster).

These options are specified in the `master_key_scope` option of the `setmasterkey` command. They should be set at the same time that the master key is set since any such change invalidates the stored passwords.

The master key scopes are:

- 1 (Constant) - The default. The **mk.dat** file can be used wherever it is copied to.
- 2 (User) - The **mk.dat** file can only be used under the same account as the one that was used when creating it.
- 3 (Machine) - The **mk.dat** file can only be used on the same machine where it was created.

Master Key Considerations when Exporting and Importing Tasks

To be able to export tasks from one machine and then import them to another, the same master key must exist on both machines. Meaning that if you change the master key on one machine, you must also change it on the other machine (using the procedure described above).

Note Replicate enforces strict access restrictions to the **mk.dat** file. Consequently, in order to export a task, you will also need to open the command prompt as an administrator (on Windows) or the product account (Linux).

For more information on importing and exporting Replicate tasks, see Exporting Tasks.

Encrypting the User Permissions File

User permissions are stored in the following repository file:

<product_dir>\Data**GlobalRepo.sqlite**

To prevent unauthorized access of this file, you can encrypt it using the procedure described below. After you perform the procedure, the repository file will be encrypted with the AES-256 bit cipher.

Note The length of any passwords specified during the procedure must be at least 32 characters.

To encrypt the repository file:

1. Open a command prompt as administrator and change the working directory to:
   <product_dir>\bin
2. Run the following command to set the master user key:
   ```
   repuictl.exe masterukey set --password your_MasterUserPassword
   ```
   Example:
   ```
   repuictl.exe masterukey set --password ANqaGYERP3UKmGLK6UNuMqrkAGxwH8FM
   ```
3. Restart the Attunity Replicate Server service.
4. Run the following command to set the repository password:
   repuictl repository setpassword --master-user-password your_MasterUserPassword --repos`itory-password your_RepositoryPassword`
   Example:
   ```
   repuictl repository setpassword --master-user-password
   ANqaGYERP3UKmGLK6UNuMqrkAGxwH8FM --repository-password
   12345678901234567890123456789000
   ```

   Note Steps 1-4 only need to be performed the first time you want to encrypt the repository file. If you subsequently need to decrypt the repository file and then re-encrypt it, they are not required.

5. Run the following command to encrypt the repository:
   ```
   repuictl.exe repository secure --on --master-user-password your_
   MasterUserPassword
   ```
   Example:
   ```
   repuictl.exe repository secure --on --master-user-password
   ANqaGYERP3UKmGLK6UNuMqrkAGxwH8FM
   ```
6. Restart the Attunity Replicate Server service.

To disable encryption for the repository:

» Run the following command:
```
repuictl repository secure --off --master-user-password your_
MasterUserPassword
```

For information on setting user permission, see User Permissions.

Securing Connections to Endpoints

Attunity Replicate communicates with the source and target endpoints (typically databases) using either the vendor provided client package or via a standard ODBC driver.

Attunity does not implement the network protocol (see important exceptions below) and for this reason, Attunity Replicate generally relies on the vendor of the source or target

endpoint to offer encryption. When setting up endpoint connections, the user is able to specify any connection properties required for the use of encryption; these properties are, invariably, vendor-specific properties. In some cases, use of encryption requires system-level settings (such as adding a certificate to the machine trust store) which are outside of the scope of Attunity Replicate. Users are referred to the operation manual of the source or target endpoint for details on how to set up encrypted client communication with the server.

One exception to the previous paragraph is for endpoints based on the Attunity Connect product (endpoints on zOS, iSeries, HP NonStop and HP OpenVMS). In this case, the network encryption is implemented by the Attunity Connect product and is based on the AES encryption algorithm with a 256-bit key.

Another exception is for endpoints that work over HTTP. In these cases, the user is advised to ensure that the endpoint server is configured to offer HTTPS and then use the appropriate https:// based URL when setting up the connection.

Application Security

As a leading provider of enterprise-class big data management solutions, Attunity understands that application security is of paramount importance. With the integration of Static Code Analysis into the development lifecycle, the code is rigorously tested for vulnerabilities before each product release.

3 | Installing Attunity Replicate

This section describes how to prepare your system for Attunity Replicate, how to install Attunity Replicate and how to access the Attunity Replicate Console.

Important: To work, Attunity Replicate needs to be set up with the proper security configuration. It is therefore strongly recommended to review Security Considerations before using the product for the first time.

In this chapter:

Installation Prerequisites

This section describes how to prepare your system to use Attunity Replicate. The requirements differ according to the platform on which you want to install Attunity Replicate and according to the desired Attunity Replicate UI Server configuration. For more information on the available UI Server configurations, see Attunity Replicate UI Server Configurations.

Software Requirements

This section describes what software is required to work with Attunity Replicate.

Windows Software Requirements

To install the Attunity Replicate Server and Console on a Windows computer, you must have the following installed on your system:

- .NET Framework 4.5.2 or above
- Visual C++ Redistributable for Visual Studio 2015. If it is not installed or if an older

version is installed, it will be installed automatically during installation.

- For a list of supported browsers, see Supported Browsers.

  For a list of supported operating systems, see Supported Windows Platforms.

Linux Software Requirements

For a list of supported Linux operating systems, see Supported Linux Platforms.

Windows Permissions

Attunity Replicate needs to be installed as an Administrator.

The following privileges are required to start the Attunity Replicate **UI Server** service (which is run as a local system service), but are dropped as soon as the service is started:

- SE_CREATE_GLOBAL_NAME
- SE_CREATE_PAGEFILE_NAME
- SE_CREATE_PERMANENT_NAME
- SE_CREATE_SYMBOLIC_LINK_NAME
- SE_CREATE_TOKEN_NAME
- SE_DEBUG_NAME
- SE_ENABLE_DELEGATION_NAME
- SE_IMPERSONATE_NAME
- SE_INC_BASE_PRIORITY_NAME
- SE_INCREASE_QUOTA_NAME
- SE_INC_WORKING_SET_NAME
- SE_LOAD_DRIVER_NAME
- SE_LOCK_MEMORY_NAME
- SE_MACHINE_ACCOUNT_NAME
- SE_MANAGE_VOLUME_NAME
- SE_PROF_SINGLE_PROCESS_NAME
- SE_RELABEL_NAME
- SE_REMOTE_SHUTDOWN_NAME
- SE_RESTORE_NAME
- SE_SECURITY_NAME
- SE_SHUTDOWN_NAME
- SE_SYNC_AGENT_NAME
- SE_SYSTEM_ENVIRONMENT_NAME
- SE_SYSTEM_PROFILE_NAME
- SE_SYSTEMTIME_NAME
- SE_TAKE_OWNERSHIP_NAME
- SE_TCB_NAME

- SE_TIME_ZONE_NAME
- SE_TRUSTED_CREDMAN_ACCESS_NAME
- SE_UNDOCK_NAME

In addition, the account that runs Replicate needs to be granted access to the **Data** directory (~\Attunity\Replicate\Data) as well as any directory containing files (such as CSV files) that need to be used in a replication task.

Recommended Hardware Configuration

This section describes the recommended hardware configurations for using Attunity Replicate. For information on the software requirements for using Attunity Replicate, see Software Requirements.

The following table describes the recommended hardware configuration for installing Attunity Replicate on Windows and Linux operating systems. Note that the recommendations apply to mid-scale systems (i.e. hundreds of tasks) rather than large-scale systems (i.e. thousands of tasks).

Table 3.1 | Recommended Hardware Configuration

	Basic System	Large System	Extra-Large System	Notes:
Processor	Quad core	Quad core base Dual-core per task	8-core base Quad core per task	Additional cores are useful in any of the following situations: » Many tasks running in parallel » Full-load performance priority » Multiple full-load processes running in parallel
Memory	8 GB	32 GB	64 GB	More memory is useful in any of the following situations: » Many tasks running in parallel » Long-running transactions on the source endpoint (for example, monthly batch processing) » Many active users on the source system
Disk requirements	320 GB 7200 RPM	500 GB 10,000 RPM RAID	500 GB 15,000 RPM RAID	A faster disk is useful in any of the following situations: » Using a file-based target, such as Greenplum or Actian Vector » Long-running source transactions that may not fit into memory » Using tasks that are set up to continue

Table 3.1 | Recommended Hardware Configuration (Cont.)

	Basic System	Large System	Extra-Large System	Notes:
				processing during target outage A larger disk is required in any of the following situations: » Using tasks that are set up to continue processing during target outage » Very large source transactions that do not fit into memory RAID is recommended for system recoverability in case of disk failure for all configurations.
Network	1 Gb	10 Gb	Two 10 Gb	

Supported Endpoints

To replicate data using Attunity Replicate, you must be sure to have a supported version of the endpoint you are working with available. For information about the endpoints you can use with Attunity Replicate, see Supported Platforms and Endpoints .

Installing Attunity Replicate

You can install Attunity Replicate on either Windows or Linux platforms. This section describes the following:

» Attunity Replicate on Windows: Installing, Upgrading and Uninstalling

» Attunity Replicate on Linux: Installing, Upgrading and Uninstalling

For information about recommended hardware requirements for working with Attunity Replicate, see Recommended Hardware Configuration.

Attunity Replicate on Windows: Installing, Upgrading and Uninstalling

Install Attunity Replicate using the `AttunityReplicate_version_X64.exe` installation kit. This kit runs on Windows 64-bit (x64) environments. For a list of the Windows versions supported by Attunity Replicate, see Windows Software Requirements.

Follow the instructions in the Setup wizard to install Attunity Replicate. Before upgrading, it is strongly recommended to back up the Replicate "Data" folder.

Later, if you need to start or stop the Attunity Replicate Server, see the following section: Starting and Stopping the Attunity Replicate Server on Windows

Note In the setup wizard's **Replication Server Location** screen, one of the options is **Connect to a remote Linux Attunity Replicate Server**. You should only select this option if you have already installed Attunity Replicate Server on a Linux machine. If you select this option, you will be prompted for the IP address and port number of the Linux machine in the following screen.
For more information on installing Attunity Replicate Server on Linux, see Attunity Replicate on Linux: Installing, Upgrading and Uninstalling.

For information on the possible deployment configurations, see Attunity Replicate UI Server Configurations.

All of the data that is created when you use Attunity Replicate is stored in a directory called **data**. By default, this directory is located in the installation directory where you install Attunity Replicate. If you want to create the data directory in a different location, select this option in the installation wizard.

If you select this option, all command line actions must be prefixed with `repctl -d <path to the data directory>`

Starting and Stopping the Attunity Replicate Server on Windows

In some cases you may need to stop and start the Attunity Replicate Server. You must do this from the Windows computer where Attunity Replicate is installed.

To stop and start the Attunity Replicate Server on Windows

» From the **Start** menu on the Windows computer where Attunity Replicate is installed, find **Attunity Replicate**; then select either **Stop Attunity Replicate Server** or **Start Attunity Replicate Server**.

Silently Installing Attunity Replicate

Attunity Replicate can be installed silently (i.e. without requiring user interaction). This option is useful, for example, if you need to install Attunity Replicate on several machines throughout your organization.

Note Before commencing the installation, make sure that the prerequisites have been met.

The installation process consists of two stages:

1. Creating a Response File
2. Running the Silent Install

Creating a Response File

Before starting the installation, you need to create a response file.

To create the response file

1. From the directory containing the Attunity Replicate setup file, run the following command (note that this will also install Attunity Replicate):

   ```
   AttunityReplicate_version_X64.exe /r /f1<my_response_file>
   ```

   where:

   `<my_response_file>` is the full path to the generated response file.

   Example:

   ```
   AttunityReplicate_version_X64.exe /r /f1C:\Replicate_install.iss
   ```

2. To change the default installation directory, open the response file in a text editor and edit the *first* **szDir** value as necessary.
3. To change the default data directory, edit the *second* **szDir** value as necessary.
4. Save the file as **<name>.iss**, e.g. **silent_inst_64.iss**.

Running the Silent Install

To silently install Attunity Replicate, open a command prompt and change the working directory to the directory containing the Attunity Replicate setup file. Then issue the following command (where <response file> is the path to the response file you created earlier):

Syntax:

```
<Replicate_setup_file> /s /f1<my_response_file> [/f2<LOG_FILE>]
```

Example:

```
C:\>AttunityReplicate_version_X64.exe /s /f1C:\temp\1\Replicate_install.iss
/f2C:\temp\1\silent_x64_install.log
```

If the installation was successful, the log file should contain the following rows:

```
[ResponseResult]
ResultCode=0
```

Silently Upgrading Attunity Replicate

Silently upgrading Attunity Replicate consists of two stages:

1. Creating a Response File
2. Running a Silent Upgrade

Creating a Response File

Before starting the upgrade, you need to create a response file.

For instructions, see Step 1 of Creating a Response File.

Running a Silent Upgrade

Before upgrading it is strongly recommended to back up the Replicate "Data" folder. To silently upgrade Attunity Replicate, open a command prompt and change the working

directory to the directory containing the Attunity Replicate setup file.

Then issue the following command (where `<my_response_file>` is the path to the response file you created earlier):

Syntax:

```
<REPLICATE_KIT> /s /f1<my_response_file> [/f2<LOG_FILE>]
```

Example:

```
C:\>AttunityReplicate_version_X64.exe /s /f1C:\temp\1\Replicate_upgrade.iss
/f2C:\temp\1\silent_x64_up.log
```

If the upgrade was successful, the log file should contain the following rows:

```
[ResponseResult]
ResultCode=0
```

Silently Uninstalling Attunity Replicate

Silently uninstalling Attunity Replicate consists of two stages:

1. Creating a Response File
2. Running a Silent Uninstall

Creating a Response File

Before starting the uninstall, you need to create a response file.

To create the response file

1. Copy the response file text below into a text editor.

   Response file text:

   ```
   [{9C614355-28A0-4C2A-98DF-DB9FD674826F}-DlgOrder]
   Dlg0={9C614355-28A0-4C2A-98DF-DB9FD674826F}-SdWelcomeMaint-0
   Count=3
   Dlg1={9C614355-28A0-4C2A-98DF-DB9FD674826F}-MessageBox-0
   Dlg2={9C614355-28A0-4C2A-98DF-DB9FD674826F}-SdFinish-0
   [{9C614355-28A0-4C2A-98DF-DB9FD674826F}-SdWelcomeMaint-0]
   Result=303
   [{9C614355-28A0-4C2A-98DF-DB9FD674826F}-MessageBox-0]
   Result=6
   [{9C614355-28A0-4C2A-98DF-DB9FD674826F}-SdFinish-0]
   Result=1
   bOpt1=0
   bOpt2=0
   ```

2. Save the file as **`<name>.iss`**, e.g. **`silent_uninst_64.iss`**.

Running a Silent Uninstall

To silently uninstall Attunity Replicate, open a command prompt and issue the following command (where *RESPONSE_FILE* is the path to the response file you created earlier and *LOG_FILE* is the path to the uninstall log file):

Syntax:

```
"C:\Program Files (x86)\InstallShield Installation Information\<directory_
containing_replicate_setup_file>\setup.exe" /s /f1RESPONSE_FILE /f2LOG_FILE
```

Note The directory containing the Replicate setup file always ends with the following string: `DB9FD674826F`

Example:

```
C:\>"C:\Program Files (x86)\InstallShield Installation Information\
{9C614350-4C2A-98DF-DB9FD674826F}\setup.exe" /s /f1C:\temp\response.iss
/f2C:\temp\1\silent_uninstall.log
```

If the uninstall was successful, the log file should contain the following rows:

```
[ResponseResult]
ResultCode=0
```

Changing the Data Directory Location on Windows

This section explains how to change the location of the Attunity Replicate Data Directory. Such a procedure may need to be performed if the drive on which the current directory resides has insufficient space or if you are moving from a temporary POC setup to production, for example.

To change the location of the data directory

1. Stop the Attunity Replicate **UI Server** and Attunity Replicate **Server** services.
2. Move the data directory to a new location. For example:
   C:\Program Files\Attunity\Replicate**Data2**
3. Open the Registry and perform the following procedure:
   a. Browse to:
      HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\services\AttunityReplicateConsole
   b. Modify the **ImagePath** string as follows:
      "C:\Program Files\Attunity\Replicate\bin\RepUiCtl.exe" -d "C:\Program Files\Attunity\Replicate**Data2**" service run
   c. Browse to:
      HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\services\AttunityReplicateServer
   d. Open the **ImagePath** string and add **-d <path_for_new_data_directory>** after

the repctl.exe path. For example:

"C:\Program Files\Attunity\Replicate\bin\repctl.exe" -d "C:\Program Files\Attunity\Replicate**Data2**" service start name=Server address=127.0.0.1 port=3552

4. Start the Attunity Replicate services.

Attunity Replicate on Linux: Installing, Upgrading and Uninstalling

This section describes how to install the Attunity Replicate Server on Linux. For information on supported Linux platforms, see Installation Prerequisites. This section contains the following:

- Installation Procedures
- Attunity Replicate Server Procedures
- Upgrading Attunity Replicate
- Uninstalling Attunity Replicate
- Changing the Data Directory Location on Linux

Note Before beginning the installation, make sure that the home directory exists. If it does not exist, the installation will fail.

Note The commands for installing, upgrading and uninstalling Attunity Replicate must be run as root or using the sudo command.

All of the commands and examples in this section assume that Attunity Replicate is being installed/upgraded/uninstalled as root. When using sudo, the parameters that you want to set need to be run as separate commands before the actual install/upgrade/uninstall command.

For example:

```
user=mike
group=sales
sudo rpm -ivh areplicate-5.0.0-135.x86_64.rpm
```

Installation Procedures

Before you carry out the following procedures, copy the Attunity Replicate RPM file to any location on the Linux computer.

The default installation directory for the Attunity Replicate is `/opt/attunity/`replicate. You can choose to install to a different directory as described below.

Note When installing or upgrading Replicate on SUSE Linux, the `--nodeps` parameter must be added to the command, as in the following example:

```
rpm -ivh --nodeps areplicate--.x86_64.rpm
```

To install Attunity Replicate in the default directory:

Issue the following single command:

```
[user=user] [group=group] [verbose=true|debug=true] rpm -ivh areplicate--.x86_64.rpm
```

Example:

```
user=mike group=sales verbose=true rpm -ivh areplicate-5.0.0-135.x86_64.rpm
```

To install Attunity Replicate in a non-default directory:

Issue the following single command:

```
[user=user] [group=group] [verbose=true|debug=true] rpm -ivh --prefix dirname areplicate-5.0.0-135.x86_64.rpm [--nodeps]
```

Example:

```
user=mike group=sales verbose=true rpm -ivh --prefix /opt/mydir/ areplicate-5.0.0-135.x86_64.rpm
```

The optional command parameters are described in the table below.

Table 3.2 | Optional Command Parameters

Parameter	Description
`[user=user]`	The default user under which Attunity Replicate is installed is attunity. You can chose to install the product under a different user by prefixing `user=user` to the command.
`[group=group]`	The default group under which Attunity Replicate is installed is attunity. You can chose to install the product under a different group by prefixing `group=group` to the command.
`[--prefix dirname]`	Prefixes the application directory with the specified *dirname*. Only required when installing Attunity Replicate to a non-default path.
`[verbose=true|debug=true]`	Specify `verbose=true` for more information during the installation or `debug=true` for detailed debug messages during the installation.
[--nodeps]	This parameter is only required when installingReplicate on Linux SUSE.

Setup will perform the following actions:

- Automatically create a new user and group named "attunity" (unless you chose to use a different user and group or a user named "attunity" already exists).
- Change the Attunity Replicate installation directory owner to the "attunity" user and group or to your preferred user and group.
- Install the required files.
- Start the service.

Attunity Replicate Server Procedures

This section describes how to verify that the Attunity Replicate Server is running as well as how to start and stop the Attunity Replicate Server.

Verifying that the Attunity Replicate Server is Running

To verify that the Attunity Replicate Server is running, issue the following command:

```
ps -ef | grep repctl
```

A list similar to the following is displayed:

```
attunity 5071 1 0 17:32 pts/0 00:00:00 /opt/attunity/replicate/bin/...
attunity 5072 5071 0 17:32 ? 00:00:00 /opt/attunity/replicate/bin/...
root 5075 4736 0 17:32 pts/0 00:00:00 grep repctl
```

Starting the Attunity Replicate Server Process

To start the Attunity Replicate Server service, run the following command (shown using the default installation path):

```
/opt/attunity/replicate/bin/arep.ctl start
```

The output will be similar to:

```
Attunity Replicate server was started as PID 5100
```

For an explanation of how to verify that the Attunity Replicate Server service is running, see Verifying that the Attunity Replicate Server is Running.

Stopping the Attunity Replicate Server Processes

To stop the Attunity Replicate Server service, run the following command (shown using the default installation path):

```
/opt/attunity/replicate/bin/arep.ctl stop
```

The output will be similar to this:

```
Attunity Replicate server was sent a stop signal
Waiting for Attunity Replicate server to stop (2 seconds)
Waiting for Attunity Replicate server to stop (17 seconds)
Attunity Replicate server is no longer running
[service command] Succeeded
```

Upgrading Attunity Replicate

This section explains how to upgrade an installation of Attunity Replicate. Before upgrading it is strongly recommended to back up the Replicate "Data" folder.

Note If the new configuration file is different than the existing configuration file, and the existing configuration file has been modified, the following warning will be displayed during the upgrade:

```
/opt/attunity/replicate/bin/repctl.cfg will be created as
/opt/attunity/replicate/bin/repctl.cfg.rpmnew
```

For more information and instructions on merging the two files, see Resolving Configuration File Conflicts.

Note When upgrading Replicate on SUSE Linux, the `--nodeps` parameter must be added to the command, as in the following example:

```
rpm -U[vh] --nodeps areplicate--.x86_64.rpm
```

To upgrade Attunity Replicate when installed in the default directory

- Issue the following command:

  `[user=username] [group=groupname] rpm -U[vh] areplicate--.x86_64.rpm`

To upgrade Attunity Replicate when installed in a non-default directory

- Issue the following command:

  `[user=name] [group=group] [nocredentials=true] rpm -U[vh] --prefix dirnameareplicate--.x86_64.rpm`

You can upgrade as a different user using the `user=name` and `group=group` parameters. You can also enable additional verbosity by prefixing the command with `verbose=true` or `debug=true`.

Note If the new or existing user and/or group is not defined locally (e.g. in Active Directory), you must include the `nocredentials=true` parameter in the command. Otherwise, the upgrade will fail.

For more information on the available command parameters, see the table Optional Command Parameters.

Resolving Configuration File Conflicts

During an upgrade, if the new version's **repctl.cfg** file contains different parameters than the existing file, and the existing file has been modified, you need to manually merge the new configuration file with the existing one.

In such a scenario, the upgrade will perform the following operations:

- The file from the new version will be renamed **repctl.cfg.rpmnew** and installed in the same directory as the **repctl.cfg** file.

  The default directory is:

  `/opt/attunity/replicate/bin/`

- The following warning will be issued:

```
[root@bldlinux-rh62 tmp]# rpm -Uvh areplicate-version-build.x86_64.rpm

Preparing... ########################################### [100%]
1:areplicate warning:
/opt/attunity/replicate/bin/repctl.cfg created as
/opt/attunity/replicate/bin/repctl.cfg.rpmnew
########################################### [100%]
```

Note that when the configuration files need to be merged, the service will not be restarted automatically. You need to restart the service manually after merging the files.

To complete the upgrade

1. Manually (and cautiously) merge the new parameters in the **repctl.cfg.rpmnew** file with the existing parameters in the **repctl.cfg** file. Save the **repctl.cfg** file.
2. Delete the **repctl.cfg.rpmnew** file by issuing the following command:

   `rm -f /opt/attunity/replicate/bin/repctl.cfg.rpmnew`
3. Restart the service by issuing the following commands from the /opt/attunity/replicate/bin directory:

```
./arep.ctl configure
./arep.ctl start
```

Uninstalling Attunity Replicate

To uninstall Attunity Replicate, type the following at the Linux prompt:

```
[verbose=true|debug=true] rpm -e areplicate
```

You can enable additional verbosity by prefixing the command with `verbose=true|debug=true`. For more information on these parameters, see the table Optional Command Parameters.

The output will be similar to this:

```
Attunity Replicate server was sent a stop signal
Waiting for Attunity Replicate server to stop (2 seconds)
Waiting for Attunity Replicate server to stop (17 seconds)
Attunity Replicate server is no longer running
[service command] Succeeded
```

To ensure that Attunity Replicate was removed from the computer, run the following command to list the sub-directories in the replicate directory.

```
ls /opt/attunity/replicate
```

Only the `tmp` and `data` directories should be listed.

Changing the Data Directory Location on Linux

This section explains how to change the location of the Attunity Replicate Data Directory. Such a procedure may need to be performed if the drive on which the current directory resides has insufficient space or if you are moving from a temporary POC setup to production, for example.

To change the Data Directory Location

1. Stop the repctl services on the Linux by running:
   `/opt/attunity/replicate/bin/arep.ctl stop`
2. Make sure all the repctl services have stopped by running:
   `ps -ef | grep repctl`
3. Move the data directory from current location (**/opt/attunity/replicate/data**) to your desired location.
4. Create a file named `site_arep_login.sh` in the Attunity Replicate bin directory.
5. Add the following command to the file:
   `export AREP_DATA=<new data directory path>`

   Example:

   `export AREP_DATA=/opt/sys232/repdata`
6. Start the Attunity Replicate Server (see Attunity Replicate Server Procedures).

Accessing the Attunity Replicate Console

You browse to the Attunity Replicate Console using a supported Web browser from a computer in the same network as the computer with the Attunity Replicate Server. For information on supported browsers, see Software Requirements.

You can access the Console from the **Start** menu of the computer where you installed Attunity Replicate.

To enable and control access to Attunity Replicate, you can create user roles as described in User Permissions.

To access Attunity Replicate

» Click Start and from the **All Programs** section point to Attunity Replicate and select **Attunity Replicate Console**.

Note When you connect to the Attunity Replicate Console, your browser will prompt you for a username and password. The username and password that you need to specify depends whether Replicate Server is installed on Windows or Linux.

- **Attunity Replicate Server on Windows:** Your domain username and password.
- **Attunity Replicate Server on Linux:** Either specify your PAM credentials or, if PAM is not set up in your environment, specify `admin` as your username and the Replicate Server password as your password.

For information on setting the Replicate Server password, see Security Considerations.

For information on PAM prerequisites, see Configuration 3: Replicate UI Console and Replicate Server Running on Linux.

Accessing Attunity Replicate from a Remote Computer

You can access Attunity Replicate from any computer in your network. The default URL is defined in a file called **ServiceConfiguration.xml**, which is located in the following directory:

`<product_dir>\data`

Note When the Attunity Replicate machine is located in a subdomain, the URL in the **ServiceConfiguration.xml** file will contain `localhost` instead of the machine name. In order to connect remotely, to the Attunity Replicate machine, you need to replace `localhost` with the actual machine name or IP address.

To access the Attunity Replicate Console from a remote computer, type the following address in the address bar of your Web browser:

Attunity Replicate Server on Windows:

`https://<computer name>/`AttunityReplicate

Attunity Replicate Server on Linux:

`https://<computer name>:<port>/`AttunityReplicate

Where <*computer name*> is the name or IP address of the computer where the Attunity Replicate Server is installed and `<port>` is the C UI Server port (3552 by default). For more information on the C UI Server component, see Attunity Replicate UI Server Configurations.

Note The person logged in to the computer where you are accessing the Console must be an authorized Attunity Replicate user. For more information, see User Permissions.

Attunity Replicate UI Server Configurations

You can either install Attunity Replicate on a single machine or on two separate machines. The possible configurations for installing Attunity Replicate on two separate machines are described below.

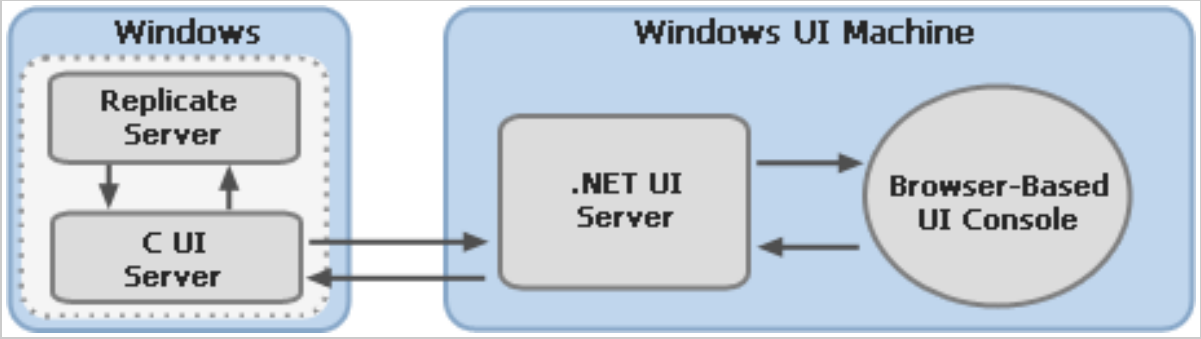
- Configuration 1: Replicate Server Running on Windows
- Configuration 2: Replicate Server Running on Linux
- Configuration 3: Replicate UI Console and Replicate Server Running on Linux

Note When the Attunity Replicate .NET UI Server is running on one machine and the Attunity Replicate Server is running on another, the Attunity Replicate Server password must be the same on both machines. The password is used during the SSL handshake to establish a secure connection between the participating machines.

For information on setting the password, see Changing the Server Password.

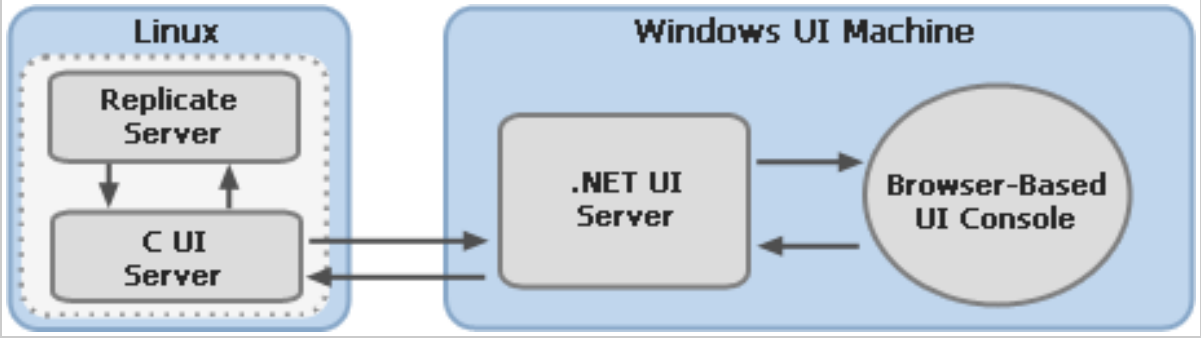
Configuration 1: Replicate Server Running on Windows

In this configuration, the Replicate Console component and the Replicate Server components are running on two separate Windows machines.



Configuration 2: Replicate Server Running on Linux

In this configuration, the Replicate Console component and the Replicate Server components are running on two separate machines - the former on Windows and the latter on Linux.



Configuration 3: Replicate UI Console and Replicate Server Running on Linux

In this configuration, the UI Console and the Web server (Attunity Replicate Server) are hosted on two separate Linux machines, though it is also possible to install them on a single machine.

Note that in such a configuration, the ability to assign different roles (as described in User Permissions) is not supported. In other words, all users will have the admin role.

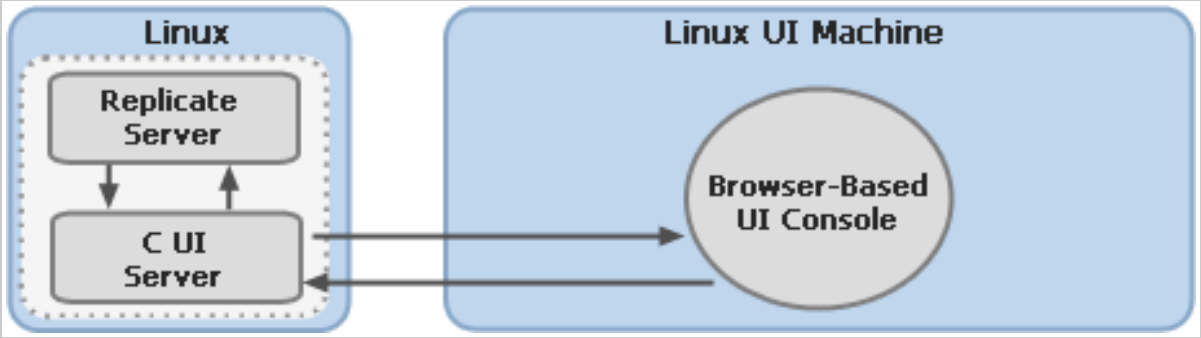
PAM Prerequisites

To establish a secure connection using PAM, make sure that the following prerequisites have been met:

- The `Attunity` user or group (or the user/group set during the installation) must be granted permission to read the file: `etc/shadow`. Note that this prerequisite is only required when Attunity Replicate is installed on two machines.
- Edit the `repctl.cfg` file and modify the path to the fully qualified name of the libpam.so.0 library if required.

  Example:

  ```
  "login_pam_libpam_full_path":"/lib64/libpam.so.0",
  "login_pam_service_name": "system-auth"
  }
  ```



Multiple Users Connecting to a Single Console

Multiple users can connect to a single Attunity Replicate Console using a Web browser, as follows:

1. Install Attunity Replicate on the computer that will serve as the Attunity Replicate Console.
2. If Attunity Replicate Server is installed on another computer (Linux for example), on the console machine, edit the Attunity Replicate URL (and port if required) in the `ServiceConfiguration.xml` file to point to that machine.

   By default, the file is located in the following directory:

   `C:\Program Files\Attunity\Replicate\data`

3. Open the Windows Services console and restart the **Attunity Replicate Console** service.
4. Connect as described in Accessing the Attunity Replicate Console above.

4 | Overview of Attunity Replicate Endpoints

Attunity Replicate lets you work with databases that already exist in your environment. There is no need to install any additional software other than Replicate. You can also use CDC Agents in Attunity Replicate Connect (ARC) as source data for replication tasks.

For a list of supported endpoint versions, see Supported Platforms and Endpoints .

In this chapter:

Supported Replicate Endpoints

Attunity Replicate can replicate data from the types of endpoints listed in the following table. A Replicate endpoint can be either a source or a target. A source endpoint contains the original data (the data you want to copy). A target endpoint is where the replicated data is stored. Source and target can be completely different endpoints.

For a list of supported source and target endpoints, see Supported Platforms and Endpoints .

You can also use CDC Agents in the Attunity Integration Suite as a source endpoint. For a list of supported ARC CDC Agents and information on how to use them with Replicate, see Using ARC CDC Agents as Endpoints.

Using ARC CDC Agents as Endpoints

In a replication project, you can use both relational and non-relational endpoints supported by Attunity Replicate Connect (ARC).

Note ARC CDC Agents can be used for capturing changes (CDC) only.

Table 4.1 | Endpoints Supported by ARC

Relational Endpoints	Non-Relational Endpoints
IBM DB2 for iSeries	HP NonStop Enscribe
IBM DB2 on z/OS	RMS
SQL/MP	VSAM
	IBM IMS

For information on how to work with ARC, see Using ARC CDC Solutions in Attunity Replicate .

Replicate Data Types

Attunity Replicate converts source data to its own data type. For data that Replicate cannot convert, it returns an error.

To see how a data type is mapped from source to target:

- See the chapter for the source target endpoint you use. In the section on data types, see the mapping table to see the Attunity Replicate data type.
- See the chapter for the target endpoint you are use. In the section on data types, see the mapping table to see how the Replicate data type maps to the target.

For example, when replicating data from an Oracle source endpoint to a Microsoft SQL Server target end point, Replicate first converts the Oracle data type BINARY to the Replicate data type BYTES. BYTES maps to the Microsoft SQL Server data type VARBINARY (Length). For more information on how Replicate maps data types, see the chapters on the individual endpoints.

The following table describes the Attunity Replicate data types. Some data types have precision and scale information that applies to them.

Table 4.2 | Replicate Data Types

Replicate Data Types	Description
STRING	A character string
WSTRING	A double-byte character string
BOOLEAN	A Boolean value
BYTES	A binary data value
DATE	A date value: Year, Month, Day
TIME	A time value: Hour, Minutes, Seconds Only the following format is supported:

Table 4.2 | Replicate Data Types (Cont.)

Replicate Data Types	Description
	`HH:MM:SS`
DATETIME	A timestamp value: Year, Month, Day, Hour, Minute, Second, Fractional Seconds The fractional seconds have a maximum scale of 9 digits. Only the following format is supported: `YYYY:MM:DD HH:MM:SS.F(9)`
INT1	A one-byte, signed integer
INT2	A two-byte, signed integer
INT4	A four-byte, signed integer
INT8	An eight-byte, signed integer
NUMERIC	An exact numeric value with a fixed precision and scale
REAL4	A single-precision floating-point value
REAL8	A double-precision floating-point value
UINT1	A one-byte, unsigned integer
UINT2	A two-byte, unsigned integer
UINT4	A four-byte, unsigned integer
UINT8	An eight-byte, unsigned integer
BLOB	Binary Large Object
CLOB	Character Large Object
NCLOB	Native Character Large Object

For more information, see **LOB support** in **Task Settings**/Metadata.

Supported DDL Statements

Attunity Replicate automatically changes the metadata of the target table to reflect DDL statements performed on the source endpoint.

Supported DDL statements include:

- Create table
- Drop table
- Rename table
- Add column
- Drop column

- Rename column
- Change column data type

For information about supported DDL statements for a specific endpoint, see the chapter describing that endpoint. For more information about DDL settings, see Apply Changes Settings.

Limitations when Capturing DDL Changes

When capturing DDL changes, the following limitations apply:

- If you change the name of a table used in a task and then stop the task, Replicate will not capture any changes made to that table after the task is resumed.
- Reallocation of a table's Primary Key columns is not supported (and will therefore not be written to the DDL History Control Table).
- When a column's data type is changed and the (same) column is then renamed *while* the task is stopped, the DDL change will appear in the DDL History Control Table as "Drop Table" and then "Add Column" when the task is resumed. Note that the same behavior can also occur as a result of prolonged latency.
- If a table is created *while* the task is stopped and the new table matches the existing table selection pattern, the new table will be captured when the task is resumed but the DDL change will not be written to the DDL History Control Table.

5 | Using the Attunity Replicate Console

The Attunity Replicate Console is a Web-based application that runs in most browsers (for information on supported browsers, see Software Requirements). You can connect from any computer to the Replicate Server.

This section describes the elements of the Replicate Console.

In this chapter:

Tasks View

The **Tasks** view is the default view that opens when you launch Attunity Replicate for the first time, as shown in the following figure. It lists all replication tasks you have defined. You use this view to view, edit, run, and delete tasks, or to create new tasks.

This view includes the following elements:

- **Toolbar** running along the top. It includes buttons that let you create a new task, open, delete, run, or stop an existing task, configure advanced run options, and manage end-point connections. See also Setting up Tasks.
- **Tasks** already defined in the system, listed in the left pane.

  You can view tasks in:

  - **Icons** view, where each icon indicates the current state of the tasks.

    See the Task Icons table for more information.
  - **Details** view, which displays a table with additional information about each task including their current state. Note that the state icons are the same as described in the Task Icons table, but without the  part of the icon.

  To toggle between these views, you can select **Icons** or **Details** from the drop-down list in the top right of the Console.

  For information about creating a task, see Designing Tasks.

  The Console displays each open task on its own tab along the top. For more information, see Viewing Specific Tasks.

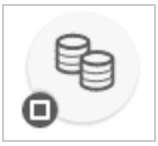
- **Endpoints map** in the right pane, which illustrates the endpoints for the task selected in the left pane. Any notifications (if defined) and log messages will be shown in the **Messages** pane below the map.
- **Messages pane** below the endpoints diagram on the right. This pane includes a **Notifications** tab for progress messages and a **Log Messages** tab for warnings and error messages issued when Replicate encounters a problem. For more information, see Reading Messages about a Task and Define the Notification Message.

To access the Tasks view

- Select **Tasks** from the drop-down list in the top left, below the Attunity Replicate logo.

The following table shows examples of task icons.

Table 5.1 | Task Icons

Task Icon	Description
	Indicates that the task has not been run yet.
	Can be one of the following: » Manually stopped by the user » Stopped due to the task definition (Full Load settings) » Stopped by the Scheduler
	Indicates that the task has stopped due to an error. When you select the task, Replicate displays a list of errors on the **Log Messages** tab at the bottom right of the console.
	Indicates that the task has stopped due to a recoverable error. When you select the task, Replicate displays a list of errors on the **Log Messages** tab at the bottom right of the console.
	Indicates that the task is running.

Viewing Specific Tasks

From the **Tasks** view, you can drill down to an individual task, provided you have already created at least one task (see Designing Tasks for more information). Two modes display different sets of information for each task:

- Designer Mode: Default mode when you open a task. Here you define endpoints, select tables, modify table settings (including filters and transformations), and create global transformation rules.
- Monitor Mode: Here you view replication task activities in real time, along with log messages and notifications.

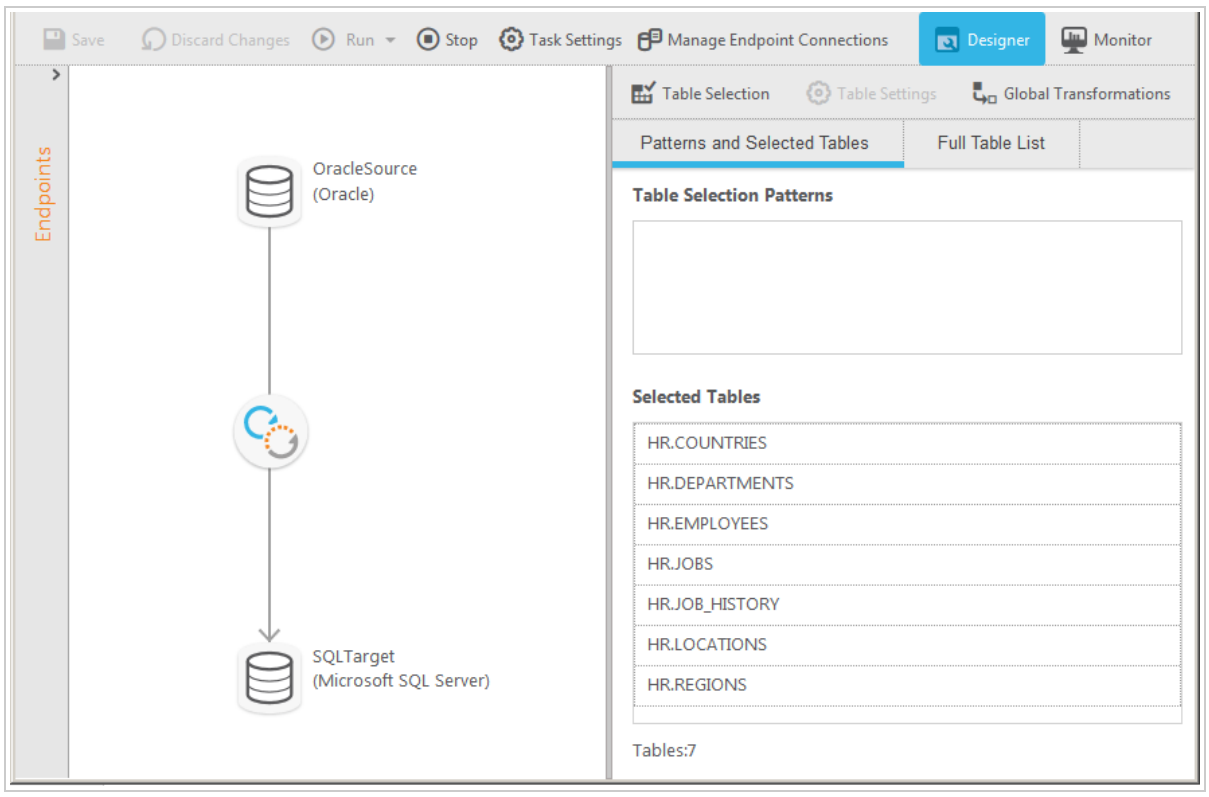
To view a specific task:

1. In the **Tasks** view, select the task you want to work with.
   The right pane displays the task diagram on the right side of the page.
2. On the **Tasks** view toolbar, click **Open**.

Designer Mode

In **Designer** mode, you define endpoints, select tables to be replicated, modify table settings (including filters and transformations), and create global transformation rules. This is the default mode when you open a task.

Figure 5.1 | Viewing a Task in Designer Mode



The **Designer** mode includes the following elements:

- **Endpoints list**: Lists the source and target endpoint connections that you added to Attunity Replicate. For more information, see Working with Endpoints. The figure shows the Endpoints List in a collapsed state, hiding the endpoints. To expand the list, click the right arrow at the top or anywhere below it. To close the panel, click the left arrow.
- **Endpoints map:** Illustrates the connection between the source and target endpoints for the task. The round icon between the endpoints represents the task type, which can indicate Full Load only, Full Load and Apply Changes, or Apply Changes only.
- When you create a task, you can drag the endpoints to the source and target drop spots as required. For more information, see Adding a Source and Target Endpoint to a Task.
- **Monitor and Designer buttons:** Lets you switch between **Monitor** mode and **Designer** mode. See also Monitor Mode and Monitoring and Controlling Replication Tasks.
- **Run button:** Lets you run the task at hand.
- **Task Settings button:** Opens the **Task Settings** dialog box. For more information, see Task Settings.
- **Manage Endpoint Connections button**: Lets you view the endpoints defined, edit them, or add new endpoints. For more information, see Working with Endpoints.
- **Select and Define Tables**: Lets you select the tables you want to include in your replication task. In addition, you can use transformation and filter operations to create new tables or to replicate parts of tables. For more information, Adding Tables and/or Views to a Task, Using Filters, and Defining Transformations for a Single Table/View.
- **Global Transformations option:** Lets you create transformations for all tables in a task. For more information, see Defining Global Transformations.

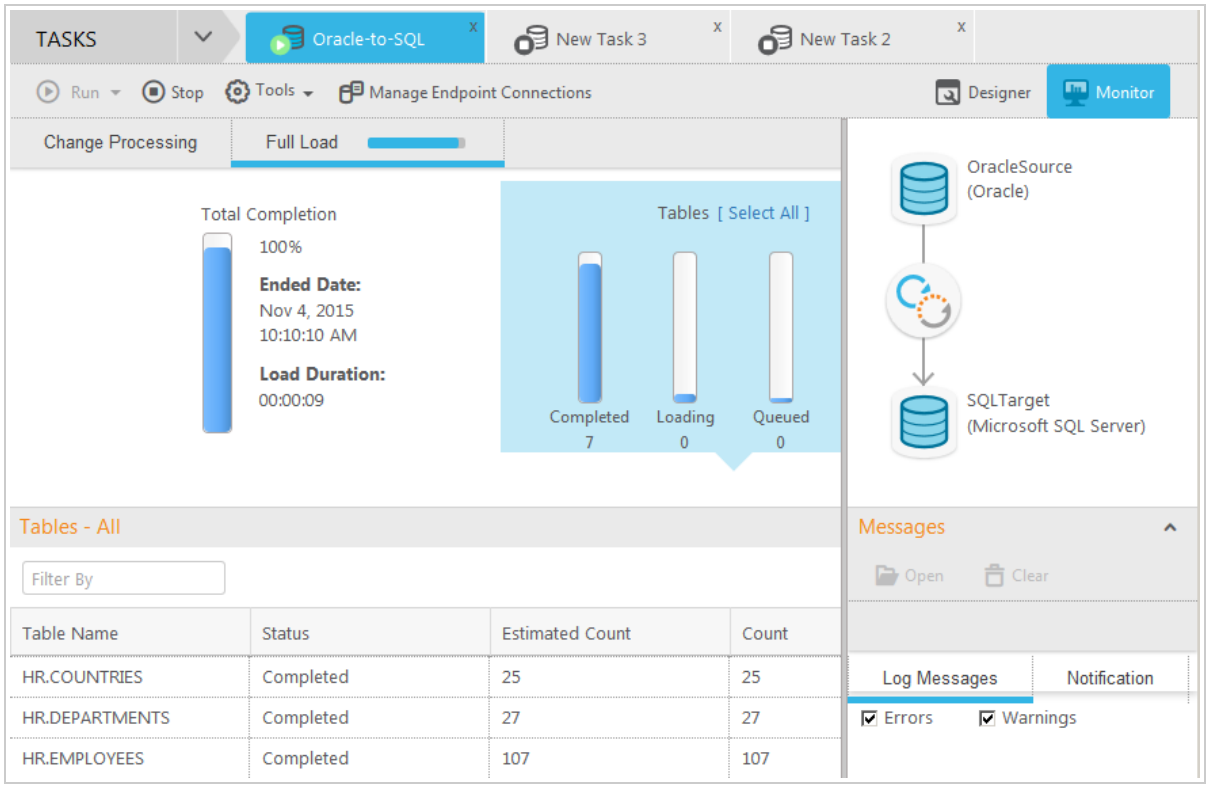
To display a task in Designer mode:

- On the right side of the toolbar, click **Designer**.

Monitor Mode

In **Monitor** mode, you view the replication task activities in real time.

Figure 5.2 | Viewing a Task in Monitor Mode



The **Monitor** mode includes the following elements:

- **Run button:** Lets you run the task at hand.
- **Manage Endpoint Connections button**: Lets you view the endpoints defined, edit them, or add new endpoints. For more information, see Working with Endpoints.
- **Monitor and Designer buttons:** Switch between **Monitor** mode and **Designer** mode. See also Monitoring and Controlling Replication Tasks, Designer Mode, Designing Tasks.
- **Tools list**: Provides access to history, log management, and status information.
- **Change Processing/Full Load tabs:** Lets you select the information you want to focus on. By default, Replicate displays the **Full Load** view (also shown in the figure).
- **Task Map:** Illustrates the connection between the source and target endpoints for the task. The round icon between the endpoints represents the task type, which can indicate Full Load only, Full Load and Apply Changes, or Apply Changes only.
- **Messages pane:** Displays notifications and logging messages. For more information, see Reading Messages about a Task.

To display a task in Monitor mode:

- On the right side of the toolbar, click **Monitor**.

Server View

The **Server** view lets you view and configure options for the Attunity Replicate Server. This view includes the following options:

- **Notifications:** Lets you create and configure notifications, configure the mail server, and create a default recipient list for sending notifications.

  See Notifications Settings for more information.
- **License:** Lets you request or register the license for Attunity Replicate.

  See License Settings for more information.
- **Global Error Handling:** Lets you set error-handling policy for the Attunity Replicate Server.

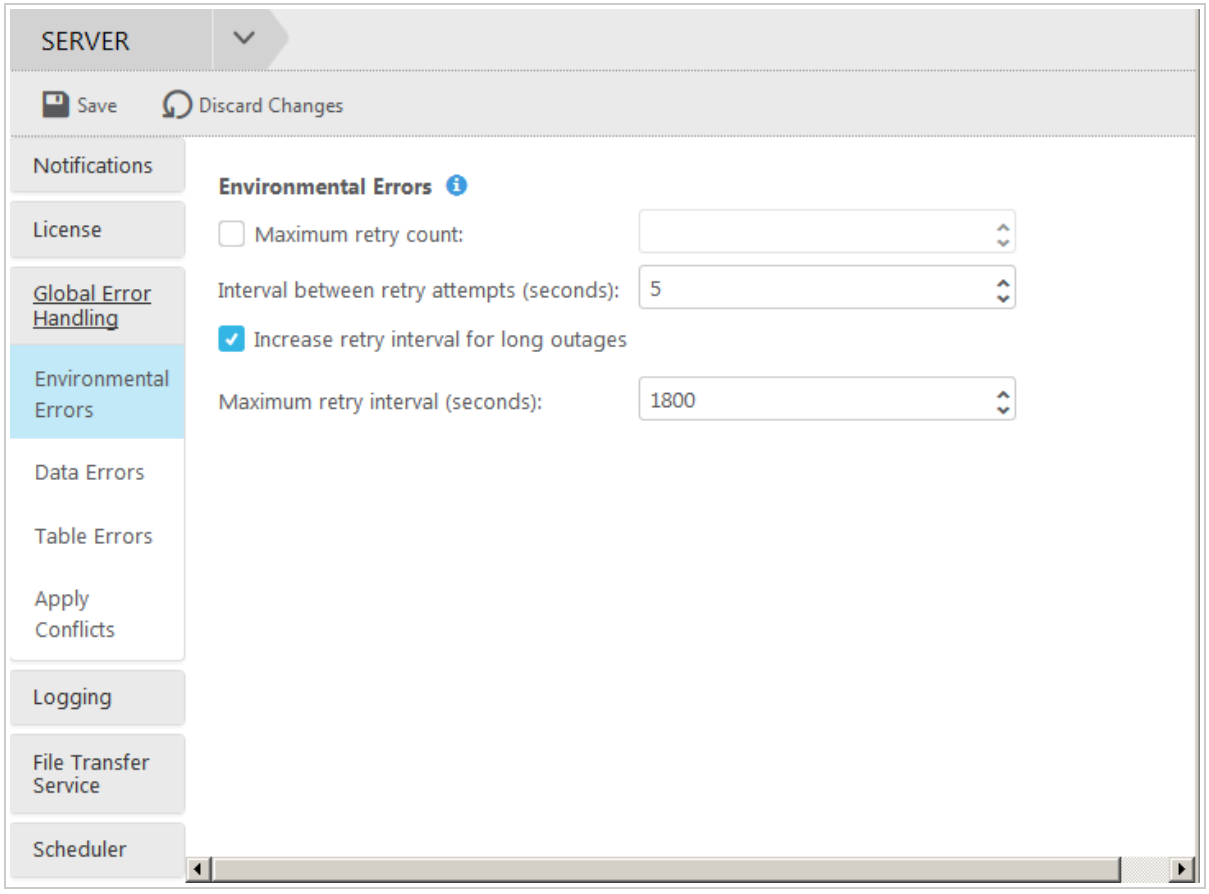
  See Global Error Handling for more information.
- **Logging:** Lets you configure the logging settings for the Attunity Replicate Server.

  See Logging for more information.

To access the server settings view:

- From the list in the top left corner, below the Attunity Replicate logo, select **Server**.

Figure 5.3 | Server View



For additional information, see Attunity Replicate Server Settings.

List Actions

The following table describes the various list actions you can perform. Note that, depending on the list type, some of the actions may not be available.

Table 5.2 | List Actions

To	Do This
Sort ascending or sort descending	Right click the desired column and select one of the sorting options as required.
Restore the default sorting order	Right click any of the column headings and select **Default Sorting**.
Export the list to a TSV file	The following lists can be exported: tasks, messages, selected tables, and processed tables (in Monitor view). Either click the **Export to TSV** button above the list or right-click any of the column headings and select **Export to TSV**. Choose where to save the file and then click **Save**.
Add or remove columns	Right click any of the column headings and select **Column Settings**. Then add or remove columns as required.
Hide a column	Right click the desired column and select **Hide Column**.

6 | Getting Started: An Attunity Replicate Tutorial

This section guides you through setting up a basic replication task for data from an Oracle source to a Microsoft SQL Server target.

In this chapter:

What You Need

For this tutorial, you need the following:

- Attunity Replicate installed on a computer in your network
- For the Oracle source:
  - Access to the HR schema tables that are part of the Oracle database installation

    Note If these tables are not available, contact your Oracle database administrator.

  - `system/<password>` for an admin user
- For the target: A Microsoft SQL Server database with the default `tempdb` system database (used to store the target tables)

  This can be installed on your local computer.
- For the Attunity Replicate Console, one of the following Internet browsers:
  - Microsoft Internet Explorer version 11 and above
  - Mozilla Firefox
  - Google Chrome

For additional installation information, see the Installation Prerequisites.

Open the Attunity Replicate Console

From the Windows Start menu, select **All Programs > Attunity Replicate > Attunity Replicate Console**.

Note You can access Attunity Replicate from any computer in your system.

To access the Console from a remote computer, type the following address in the address bar of your Web browser:

```
http://<computer name>/attunityreplicate
```

where <*computer name*> is the name or IP address of the computer (including the Windows domain name) on which the Attunity Replicate Server is installed.

Note The person logged in to the computer hosting the Console must be an authorized Attunity Replicate user. For more information, see User Permissions.

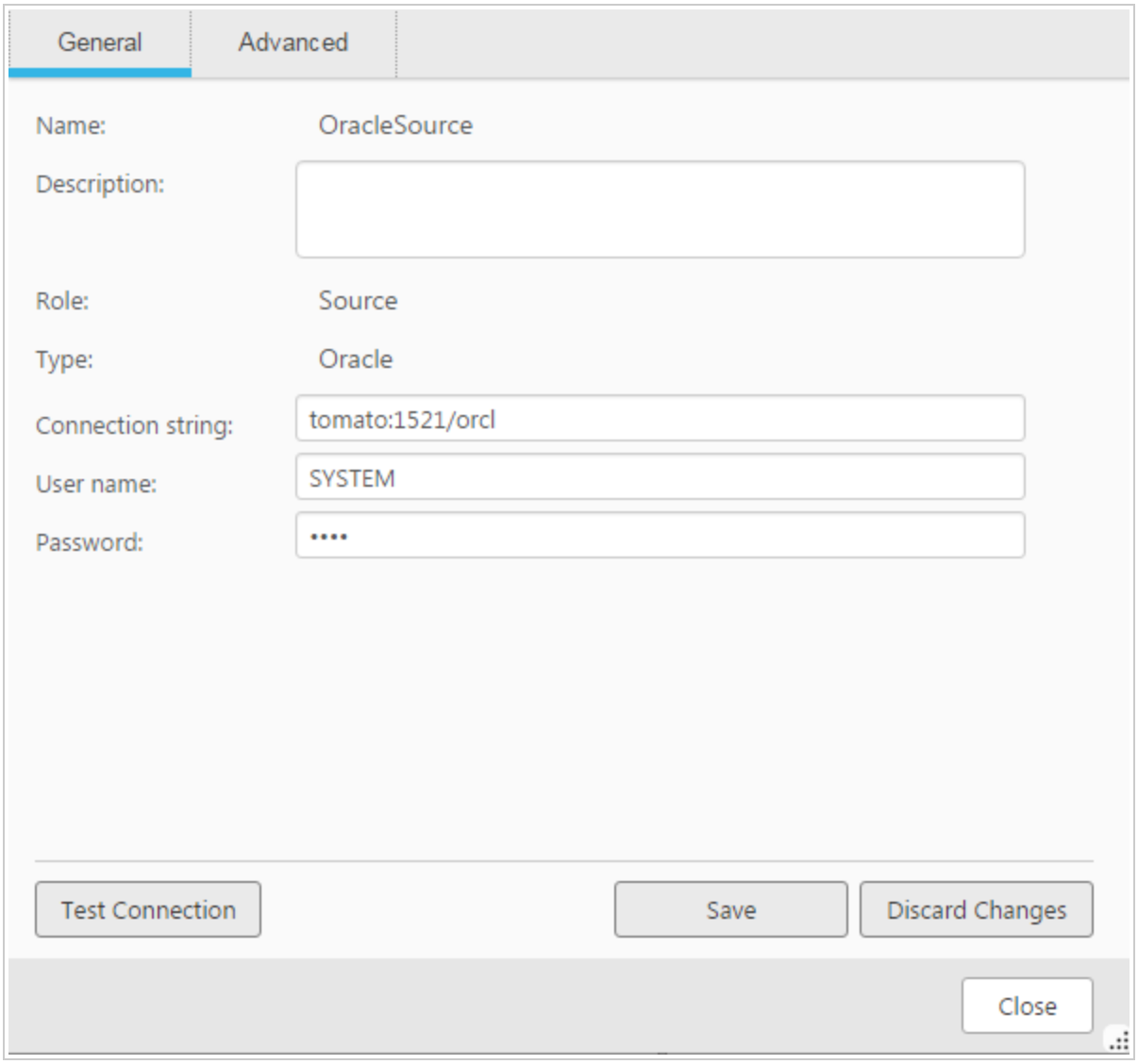
Add an Oracle database as a Source

This task guides you through adding and configuring an Oracle database as the source database connection. This is the database from where you want to replicate data.

To add an Oracle source database:

1. In Task view, click **Manage Endpoint Connections**.
   The **Manage Endpoint Connections** dialog box opens.
2. Click **New Endpoint Connection**.
3. Provide the following information:
   - **Name**: **Type OracleSource**.
   - **Description**: Optionally, enter a description or leave blank.
   - **Role**: Select **Source**.
   - **Type**: Select **Oracle**.
   - **Connection string**: Enter the connect string to the Oracle database you work with, in any Oracle format.
     For example, if you connect to an Oracle database on a computer called **tomato** using the default Oracle port and service name, the connect string looks like this:
     `tomato:1521/orcl`
   - **User Name**: Enter the user name for the Oracle database you work with.
     The default user name is **SYSTEM**.
   - **Password**: Enter the password for the Oracle database you work with.
     The default password is **manager**.

The figure below shows the data that you need to enter on the General tab of the Oracle database.



4. Click **Test Connection** to verify the information you entered and the availability of the database.
5. Click **Save** to add the database.

You can also set advanced settings for the Oracle database, but this beyond the scope of this tutorial. For more information, see Setting Advanced Connection Properties Using LogMiner.

For information on adding other types of databases, see the chapter for the required database. For a list of supported databases, see Supported Platforms and Endpoints .

Add a Microsoft SQL Server database as a Target

This task guides you through adding and configuring a Microsoft SQL Server endpoint as the target database connection. This is the database to where you want to replicate data.

To add a Microsoft SQL Server target endpoint:

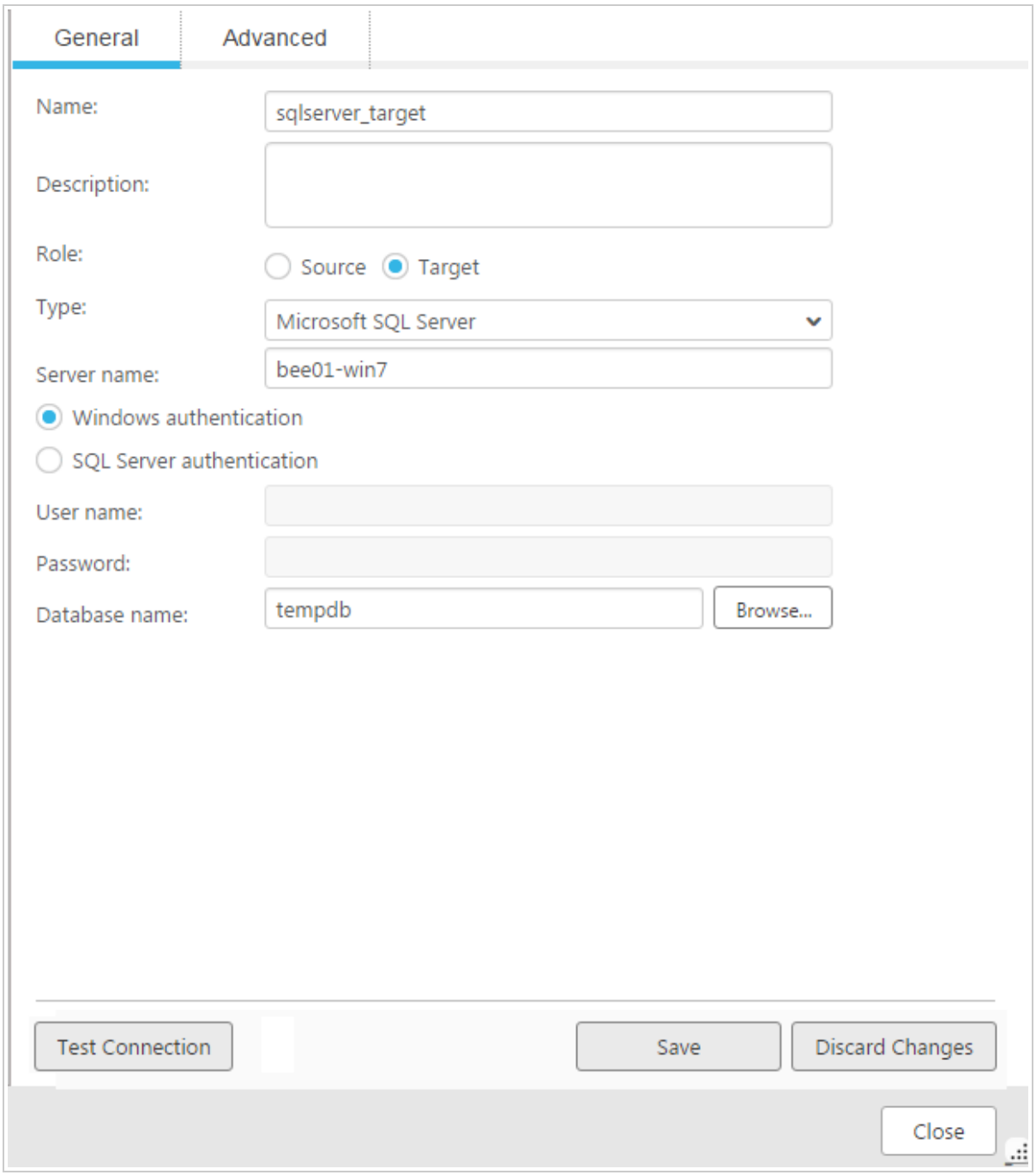
1. In Tasks view, click **Manage Endpoint Connections**.

   The Manage Endpoint Connections dialog box opens.
2. Click **New Endpoint Connection**.
3. Provide the following information:
   - **Name**: Type sqlserver_target.
   - **Description**: Optionally, enter a description or leave blank.
   - **Role**: Select **Target**.
   - **Server name**: Enter the name of the computer where your Microsoft SQL Server database is installed.

     For example, if you connect to a Microsoft SQL Server database on a computer called `bee`, enter **bee**.
   - Select one of the following:

     Windows authentication if your Microsoft SQL Server database is configured to accept Windows authentication.

     Microsoft SQL Server authentication if your Microsoft SQL Server database is not configured to accept Windows authentication. In this case, you also need to provide a valid user name and password.
   - **Database name**: Enter **tempdb**, which is the name of the database to where you are going to replicate data. If you created a new database for this purpose, enter the name of that database.



4. Click **Test Connection** to verify the information you entered and the availability of the database.
5. Click **Save** to add the database.

You can also set advanced settings for the Microsoft SQL Server database, but this is beyond the scope of this tutorial. For more information, see Setting Advanced Connection Properties.

For information on adding other types of databases, see the chapter for the required database. For a list of supported databases, see Supported Platforms and Endpoints .

Add a Replication Task

This task guides you through defining a replication task that copies the data from the HR.EMPLOYEES and HR.JOBS tables. It is not mandatory to add a source and a target database prior to this step; you can also do this as part of setting up the replication task.

By default the Oracle database includes the HR schema. You will make a copy of the same tables in your Microsoft SQL Server `tempdb`. The EMPLOYEES and JOBS tables created in Microsoft SQL Server will be identical to the Oracle tables.

For information on how to use Transformations and Filters when creating a replication task, see Defining Transformations for a Single Table/View and Using Filters.

Adding a replication task includes the following subtasks:

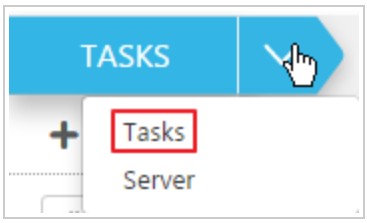
- Add a Replication Task to the Attunity Replicate Console
- Add the Source and Target databases to the Task
- Select Tables for the Replication Task

Add a Replication Task to the Attunity Replicate Console

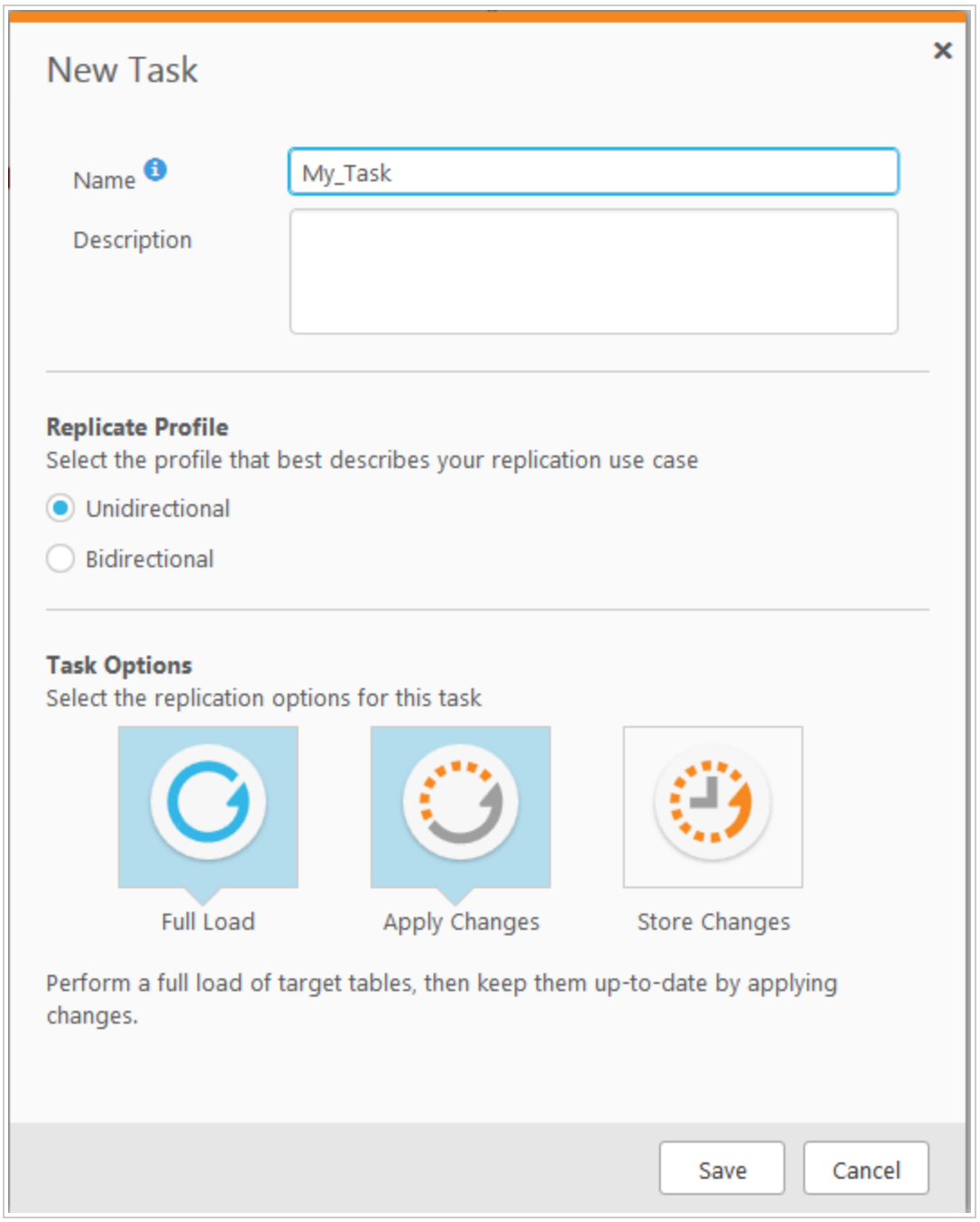
This task guides you through adding a replication task to the Attunity Replicate Console.

To add a replication task:

1. Make sure **Tasks** is selected in the upper left corner of the Attunity Replicate Console.



2. Click **New Task** to open the New Task dialog box.
3. In the New Task dialog box, in the **Name** field, type ***My_Task*** and click **OK**.



The Attunity Replicate Console displays the task on a new tab. By default, because the task has not been set up yet, the tab opens in Designer view. The diagram on the left serves as a drop-off point for the source and target databases you defined previously. The right pane lets you select the tables you want to work with and carry out transformations and filtering operations. For more information, see Tasks View, Viewing Specific Tasks, and Designing Tasks.

If needed, you can also change the default task settings. For more information, see Task Settings.

Add the Source and Target databases to the Task

This section guides you through adding the source and target endpoints to the replication task, which is a simple drag-and-drop operation. In the **Endpoints** tab, the following icons help you distinguish between source and target endpoints:

Source endpoint, which is represented by a database, file, or NoSQL icon, depending on the endpoint type, with an orange arrow pointing away from the source (a database in this example).

Target endpoint, which is represented by a database, file or NoSQL icon, depending on the endpoint type, with a blue arrow pointing toward the target (a database in this example).

The **Endpoints** pane consists of **All**, **Source**, and **Target** tabs. Because each database is named in a way that reflects whether it is a source or a target, click the **All** tab.

To add the source or target endpoints to the task:

1. In the **Endpoints** pane, click the **All** tab.
2. Drag the `OracleSource` database to the **Drop source endpoint here** area in the endpoints diagram.

3. Drag the `sqlserver_target`database to the **Drop target endpoint here** area.

Next, you can select the tables from the source database to use in the replication task. For more information, see Designing Tasks.

Select Tables for the Replication Task

After adding the source and target databases, you now need to select which Oracle source tables you want to replicate to the Microsoft SQL Server target.

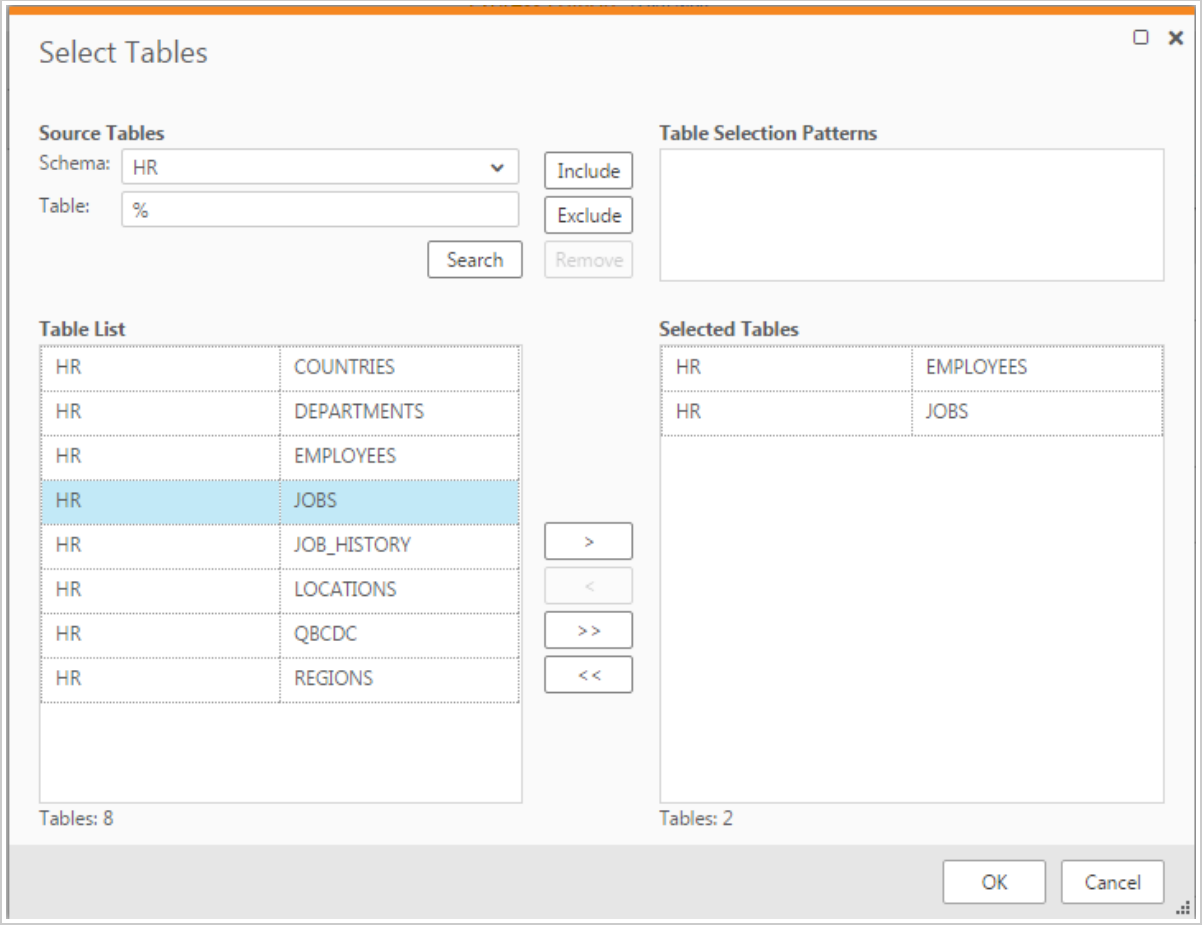
This task guides you through selecting specific tables (HR.EMPLOYEES and HR.JOBS) from the Oracle source. Replicate takes all of the data from these tables "as is" and copies it to the Microsoft SQL Server target.

Note If you need to copy only some of the data to the target database, you need to use a filter. For information, see Using Filters.

If you need to copy the data into the target using different rows or columns than those in the source, you need to use transforms. For more information, see Defining Transformations for a Single Table/View.

To add tables to the replication task:

1. In the right pane of the Attunity Replicate Console, click **Table Selection**. The Select Tables dialog box opens.
2. In the Select Tables dialog box, do the following:
   - From the **Schema** list, select **HR**, and then click **Search**.
   - From the **Table List**, select **EMPLOYEES**, and then click the right arrow to select that table.
   - Repeat these steps for the **JOBS** table.
   - Click **OK**.



3. On the task tab, click **Save**. The task is now ready to run.

Run and Monitor the Replication Task

You can now run the replication task and see the results of the replication in real time. This task guides you through running the replication task as a full load and viewing the progress in the Monitor. Additional run options are also available. For more information, see Using the Run Button Options.

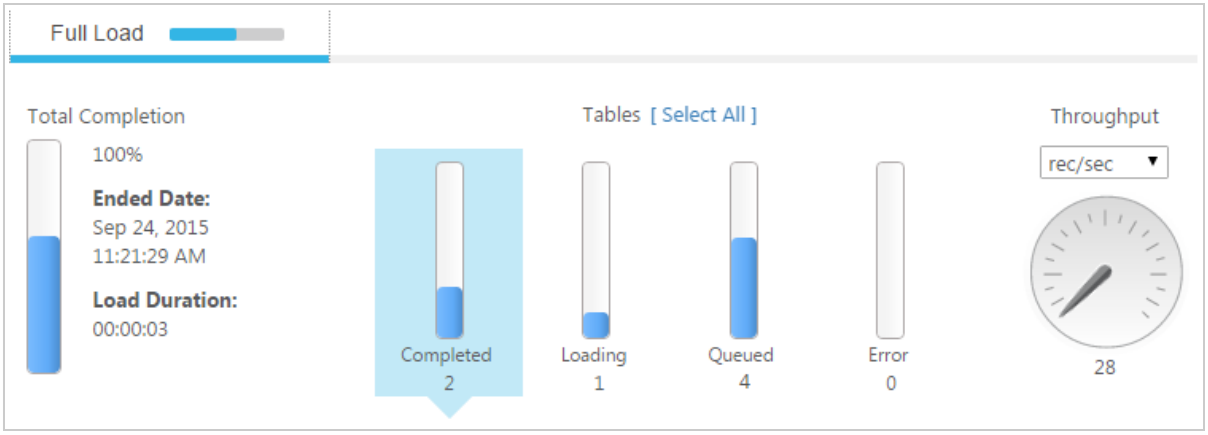
To run and monitor the replication task:

1. On the task tab, click **Run**.

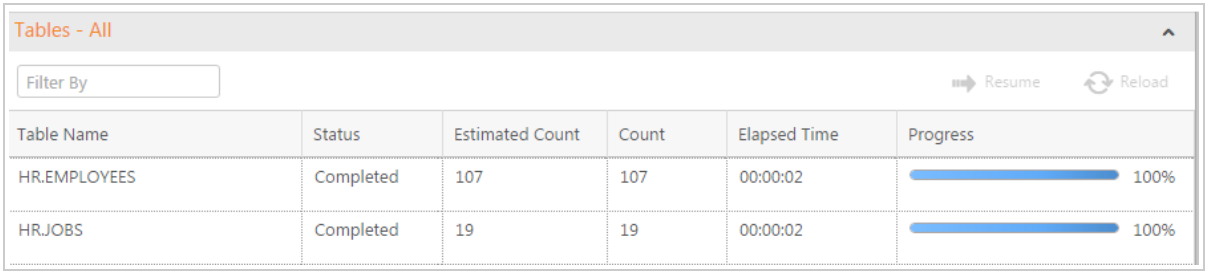
   The **Starting task** message displays, and the console switches to Monitor view, which includes gauges and graphs on two tabs:

   - **Full Load** tab: Indicates the status progress during the full load process.
   - **Change Processing** tab: Monitors changes that occur after the full load completes.

   For information on reading the data presented in these sections, see Viewing Information in the Monitor.



2. Click the **Select All** link above the Tables graphs. Replicate displays a table below the graphs with information about each of the tables being processed in the task.



3. Click the individual bar graphs, such as the Completed graph and the Loading graph, to view additional information.

   For information about the data supplied in these tables, see Monitoring Full-Load Operations.

View the Replicated Tables in Microsoft SQL Server

This task guides you through viewing the `tempdb` database in Microsoft SQL Server. You will see that this database now includes two new tables: HR.EMPLOYEES and HR.JOBS.

To view the replicated tables in Microsoft SQL Server:

1. From the Windows **Start** menu, go to **All Programs** > **Microsoft SQL Server** > **Microsoft SQL Server Management Studio**.
2. In the Object Explorer, find the Microsoft SQL Server target computer you are working with.
3. Expand the **databases** folder for that computer, then expand the **System databases** folder, then expand the **tempdb** database. The `EMPLOYEES` and `JOBS` tables should now appear in the list.
4. Right-click the `EMPLOYEES` table and select **Select Top 1000** Rows. Check that there is data in the table.
5. Right-click the `JOBS` table and select **Select Top 1000** Rows. Check that there is data in the table.

7 | Designing Tasks

This section describes how to design a replication task. To design a replication task, you must first be sure that you have configured at least one source endpoint and one target endpoint to work with Attunity Replicate.

It is also possible to customize a task by creating new tables or columns for the target endpoint or by selecting only some of the data from each column to be replicated. This is done using transformations and filters.

Note A number of variables affect the amount of tasks that can be run on a single Replicate Server, including the task configuration (e.g. how many tables are being replicated), the size of the source tables and the hardware configuration of the Replicate Server machine. Bearing this in mind, the number of tasks that can be run on a single Replicate Server should not exceed 100 (and may need to be significantly less depending on the aforementioned variables). Best practice is to perform load testing in a Test environment before moving to Production.

For more information, see:

- Customizing Tasks
- Replication Tasks

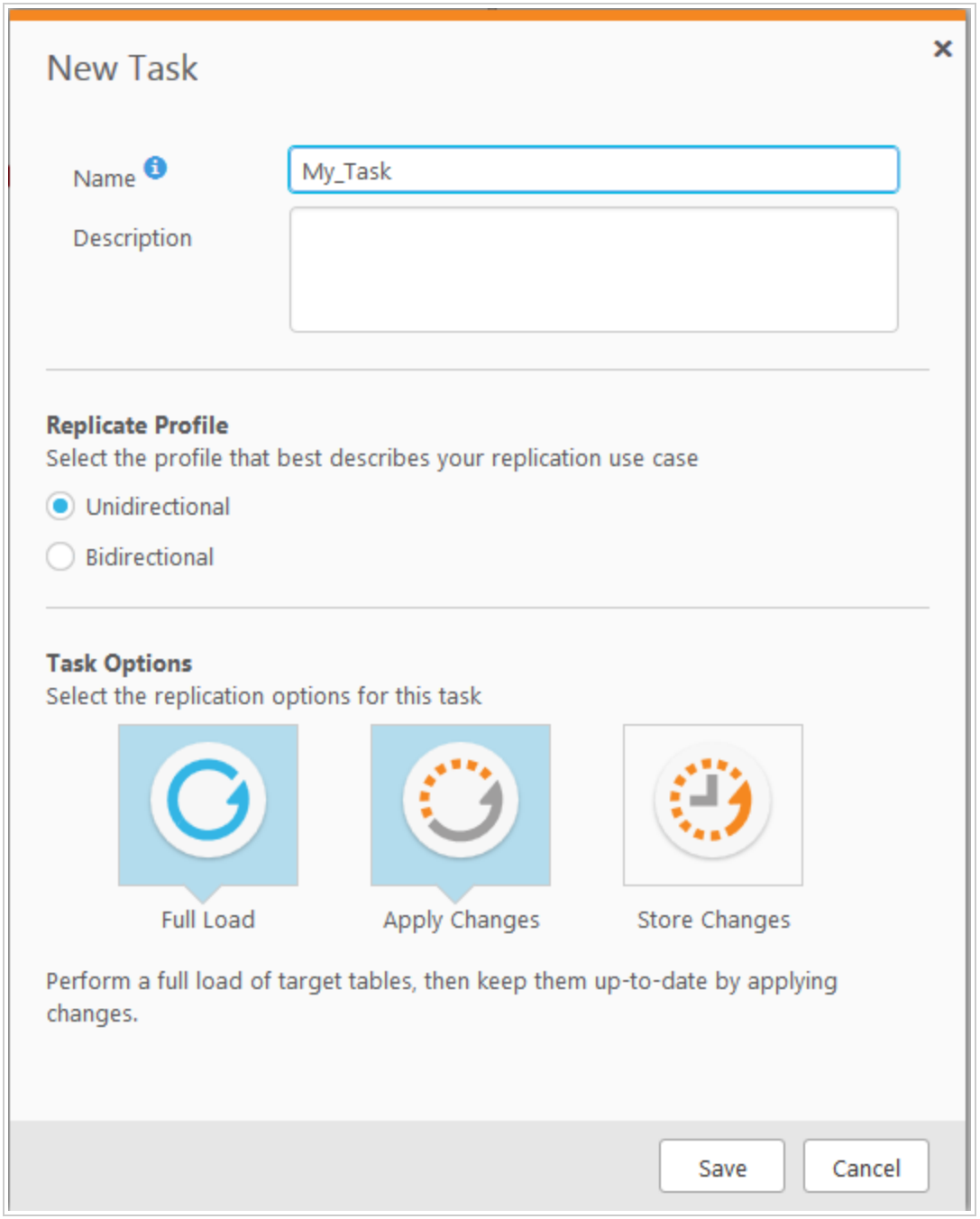
In this chapter:

Setting up Tasks

Before you get started with designing the features that you need for a task, you must define the task's default behavior.

To get started setting up a task:

1. In Tasks view, click **New Task**. The **New Task** dialog box opens.



2. Enter a name for the task. The name should be descriptive to indicate the purpose of the task. The name cannot exceed 32 characters, contain non-Latin characters, or contain any of the following characters: | \ / : * ? " < >
3. Optionally, enter a description for the task.

4. Choose one of the following replication profiles:
   - **Unidirectional** - Choose to replicate between endpoints for the purpose of data warehousing or business intelligence.
   - **Bidirectional** - Choose to synchronize records between two endpoints.

     For more information, see the instructions on setting up Bidirectional Replication.
5. Select task options:
   - **Full Load**: Click to enable or disable Full Load options for this task.

     When full load is enabled, Attunity Replicate loads the initial source data to the target endpoint. By default a full load is carried out for this task. If you want to change this setting after you begin working with this task, you make the change in the Task Settings, Full Load tab.
   - **Apply Changes**: Click to enable or disable Apply Changes (Change Processing).

     When this option is enabled, Attunity Replicate processes the changes. By default, change processing is carried out for this task. You can view the change processing in the Monitor view.

     For more information, see Monitoring Change Processing Operations. If you want to change this setting after you begin working with this task, you make the change in the Task Settings > Change Processing tab.

     Note When the **Bidirectional** replication option is selected, the **Apply Changes** option cannot be disabled.
   - **Store Changes**: Click this button to enable or disable **Store Changes**.

     If this option is enabled, changes are stored in change tables or in an audit table. By default, changes are not stored.

     For information about storing and applying changes, see Working with Change Tables and Using an Audit Table.

     Note When the **Bidirectional** replication option is selected, the **Store Changes** button will be unavailable.
6. Click **OK** to close the **New Task** dialog box and save your settings.

Bidirectional Replication

Bidirectional replication enables organizations to synchronize data between two endpoints (henceforth referred to as Endpoint A and Endpoint B), ensuring that both endpoints contain identical records. The endpoints can either be the same type (e.g. Oracle-to-Oracle) or different types (e.g. Microsoft SQL Server-to-Oracle). To implement bidirectional replication, two **Bidirectional Replication** tasks need to be defined: one that captures changes made to Endpoint A and replicates them to Endpoint B (Task 1) and another that captures changes made to Endpoint B and replicates them to Endpoint A (Task 2). An explanation of how to set up these tasks is provided below.

Limitations

The following limitations apply to Bidirectional replication tasks:

- Bidirectional replication does not currently support conflict resolution. To prevent conflicts, organizations should ensure that the application that updates the endpoints participating in a bidirectional replication task, does not simultaneously update the same record in both endpoints.

  In other words, if a record in Endpoint A was updated, the equivalent record in Endpoint B should only be updated after the update from Endpoint A is replicated to Endpoint B.
- Bidirectional replication tasks currently only support DDL statements from one source only.

  Note The `CREATE TABLE` DDL is not supported.
- To ensure that the source and target endpoints are identical, transformations and filters should not be used in bidirectional replication tasks.
- The **Use direct path full load** option in the Oracle target endpoint settings is not supported.
- The **Stop the task after Full Load completes and:** option in the task settings' Full Load Settings tab is not supported.
- The task's **Change Processing Mode** must be set to Transactional apply.

Supported Endpoints

Bidirectional tasks support the following endpoints:

Source Endpoints:

- Oracle
- Microsoft SQL Server
- MySQL
- PostgreSQL
- All AIS sources
- File Channel
- SAP Sybase ASE

Target Endpoints:

- Oracle
- Microsoft SQL Server
- MySQL
- PostgreSQL
- ODBC
- File Channel
- SAP Sybase ASE

Setting up Bidirectional Replication

This section describes how to set up a Bidirectional replication task in Attunity Replicate . Note that if Endpoint B contains tables that do not exist in Endpoint A, you must first set up and run a data warehousing or business intelligence task that replicates data from Endpoint B to Endpoint A.

To replicate data from Endpoint B to Endpoint A:

1. Define and run a **Data Warehousing or Business Intelligence** task that replicates data from Endpoint B to Endpoint A with Full Load enabled only (make sure to disable the Apply Changes option).
2. When the task completes, first verify that all the required tables exist in Endpoint A.
3. Continue with setting up Bidirectional Task 1, as described in the following procedure.

To set up Bidirectional Task 1:

1. Define a **Bidirectional Replication** task that replicates data from Endpoint A to Endpoint B.

   Note In a bidirectional replication task, **Full Load** replication is not enabled by default since it is assumed that both endpoints contain identical tables. If this is not the case (for instance, if Endpoint A contains tables that do not exist in Endpoint B), enable **Full Load** replication as well.

2. Specify a source and target **Loopback prevention table schema** in the task settings' **Loopback Prevention** tab. For more information about loopback prevention settings, see Bidirectional.
3. Run the task.

To set up Bidirectional Task 2:

1. Define another **Bidirectional Replication** task that replicates data from Endpoint B to Endpoint A.
2. Specify a source and target **Loopback prevention table schema** in the task settings' **Loopback Prevention** tab. For more information about loopback prevention settings, see Bidirectional.
3. Run the task. If Full Load was enabled when replicating data from Endpoint A to Endpoint B, you must first wait for the Full Load replication to complete before running the task.

Using Bidirectional Replication with the File Channel Endpoint

You can use bidirectional replication together with the File Channel endpoint. This is useful if you need to synchronize two endpoints that are either not able to communicate with each other (i.e. are not physically connected) or are located in the WAN. The process involves setting up six separate tasks: Two *Full Load-only* Data Warehousing or Business Intelligence tasks and four *Apply Changes-only* Bidirectional tasks.

For information on setting up the File Channel endpoint, see Using the Attunity Replicate File Channel.

To set up bidirectional replication with File Channel Endpoints

1. Set up and run two Full Load only Data Warehousing or Business Intelligence tasks.
   Example (FC = File Channel):
   Task 1: MySQL --> FC Target **Task 2:** FC Source --> Oracle
2. Wait for the *Full Load-only* tasks to finish.
3. Set up and run four *Apply Changes-only* Bidirectional tasks.
   Example (FC = File Channel):
   Task 1: MySQL Source --> FC Target **Task 2:** FC Source 1 --> Oracle Target
   Task 3: Oracle Source --> FC Target 2 **Task 4:** FC Source 2 --> MySQL Target

Working with Endpoints

Attunity Replicate requires information to connect to the source and target endpoints that you want to use in a task. For a list of endpoints you can work with in Attunity Replicate, see Supported Platforms and Endpoints .

You use the **Manage Endpoint Connections** window to add endpoints and edit and view the endpoint connection information.

Note The name cannot exceed 32 characters, contain non-Latin characters, or contain any of the following characters: | \ / : * ? " < >

- Adding an Endpoint
- Editing Endpoint Configuration Information
- Viewing Endpoint Configuration Information

Adding an Endpoint

Before you can begin to design a task, you must add endpoints to the Replicate server. To use an endpoint, you must have access to it somewhere in your system. When you add the endpoint to the Replicate server, you must provide connection information and proper user credentials.

Once you add endpoints to the Replicate server, you can begin to use them to build a replication task. For information on how to add an endpoint to a replication task, see Adding a Source and Target Endpoint to a Task.

To add an endpoint:

1. In the Tasks view, click **Manage Endpoints**.
   The **Manage Endpoint Connections** window opens.

2. In the **Manage Endpoint Connections** window, click **New Endpoint**.
3. Select the type of endpoint you are using. The information that you must enter depends on which endpoint you select.

   For more information, see the chapter that describes the endpoint you are using. For a list of supported databases, see Supported Platforms and Endpoints .

Editing Endpoint Configuration Information

After you add the endpoint to the Replicate server and provide the connection information, you can make changes to some of the information.

Note You *cannot* change the following information in the endpoint window:

- The name you provided for the endpoint.
- The endpoint **Type**, for example Oracle or Microsoft SQL Server.
- The endpoint **role**, either SOURCE or TARGET.

To edit endpoint configuration information:

1. In the **Manage Endpoint Connections** window, select the endpoint you want to edit.

   or

   In the **Endpoints** list on the left of the Designer view, double-click the endpoint you want to edit. Note that this option is only available when editing a specific task.

   The **Manage Endpoint Connections** window opens with the selected endpoint settings.
2. Make changes to the information in any of the tabs in the window.

   For more information, see the chapter for the specific Attunity Replicate endpoint you are using. For information which endpoints are supported by Attunity Replicate, see Supported Platforms and Endpoints .

Viewing Endpoint Configuration Information

After you add the endpoint to the Replicate server and provide the connection information, you can view the information in the **Manage Endpoint Connections** window.

To view endpoint configuration information:

- Select an endpoint from the **Endpoints** list in the left pane; then click the tabs to view the information.

Testing an Endpoint Connection

You can try to contact the endpoint to make sure that you are connected to the endpoint you want to work with.

To test the endpoint connection:

1. In the **Manage Endpoint Connections** window, select the endpoint you want to work with.
2. At the bottom of the endpoint's **General** tab, click **Test Connection**.

   If the connection is successful, a success message is displayed and a green check mark icon appears next to the **Test Connection** button.

   If the connection fails, an error message is displayed at the bottom of the dialog box and the **View Log** button becomes available.
3. If the connection is successful, click **Close**.

   If the connection fails, click **View Log** to view the server log entry with information for the connection failure.

Duplicating Endpoints

You can duplicate an endpoint if you need to define a new endpoint with similar settings. Except for the name, all endpoint settings are duplicated to the new endpoint.

To duplicate an endpoint:

1. In the left panel of the **Manage Endpoint Connections** window, click the endpoint you want to duplicate.
2. Click **Duplicate**.
3. On the General tab, edit the name for the endpoint.
4. Make any other necessary changes.
5. Click **Save**; then click **Close**.

Searching for Endpoints

You can search for endpoints by typing a sequence of letters in the **Filter by** box above the endpoints list. For example, to search for all endpoints whose names contain the string "Oracle", type "or". Only endpoints that match the search string are displayed.

Deleting Endpoints

You can delete endpoints that you no longer require. Note that to delete an endpoint that is defined as a source or target in a task, you first need to remove the endpoint from the task.

To delete an endpoint:

- In the left panel of the **Manage Endpoint Connections** window, Select the endpoint and click **Delete**.

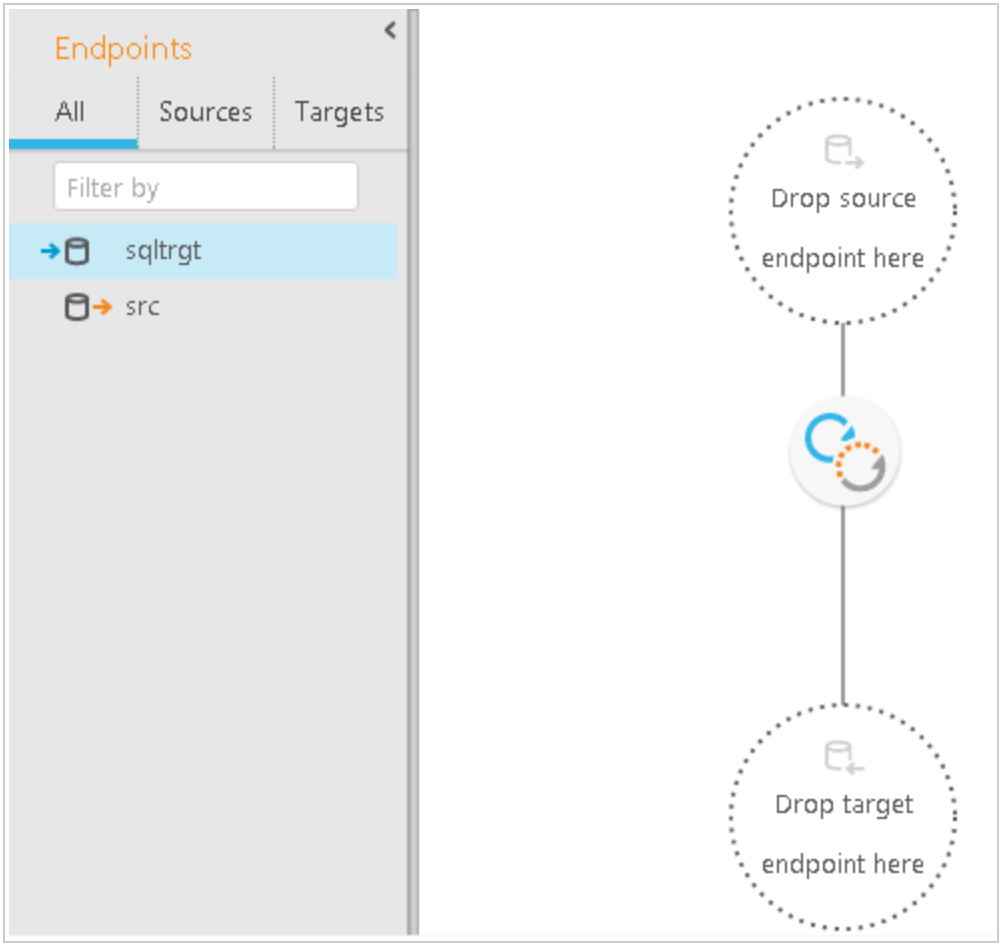
Adding a Source and Target Endpoint to a Task

Once you have added the endpoints, you can design the replication task. The first step in this process is to define the source endpoint where your data is currently stored and the target endpoints where you want to replicate the data. To do this, you just drag one of the endpoints you added into the task map (in Designer mode).

Once you select the endpoint for your task, you must select the tables from the source endpoint to be replicated. The next step in creating a replication task is Adding Tables and/or Views to a Task.

To add source and target endpoints to a task:

1. Do one of the following:
   - Create a new task. When you click **OK** in the **Create New Task** dialog box, the task opens on a dedicated tab. For more information, see Setting up Tasks.
   - In the Tasks view, select the task to which you want to add endpoints and click **Open**. The task opens on a dedicated tab.
2. The Task map is displayed, with the available endpoints listed in the pane on the left, as shown in the following figure.



3. Drag a source endpoint to the top circle in the task map (that contains the text **Drop source endpoint here**). If dragging is not possible, make sure that the endpoint you are using is defined as a source endpoint.
4. Drag a target endpoint to the bottom circle in the task map (that contains the text **Drop target endpoint here**). If dragging is not possible, make sure that the endpoint you are using is defined as a target endpoint.
5. Click **Save**.

Adding Tables and/or Views to a Task

This procedure describes how to select the source tables or views that you want to replicate. Note that tables can be selected from any supported endpoint, but views can only be selected from the **ODBC** or **ODBC with CDC** source endpoints.

Views are replicated to the target endpoint as tables.

Note When working with an ODBC with CDC source, any views and tables that you want to replicate must have the same context field(s). If you only want to replicate views, then all of the views must have the same context field(s).

For information on setting up context fields, see Setting Change Processing Parameters.

Once you have selected tables/views to replicate, you can run the replication task. However if you want to make any changes to the structure of the tables in the target endpoint or only select specific columns, you will need to carry out one or both of the following procedures:

- Defining Transformations for a Single Table/View
- Using Filters

To select tables/views:

1. Open the task you are working with if it is not already displayed in a dedicated tab.

   For information on opening a task, see Editing a Replication Task.
2. In Designer mode, on the right side, click **Table Selection**.

   If the source endpoint does not support view selection, the **Select Tables** dialog box opens. If the source endpoint supports view selection, the **Select Tables/Views** dialog box opens.

   See the following for information on how to work with the **Select Tables/Select Tables/Views** dialog box:

   - Searching for Tables/Views to use in a Replication Task
   - Selecting Specific Tables/Views for Replication
   - Creating Table/View Selection Patterns
   - Setting Load Order

Searching for Tables/Views to use in a Replication Task

This topic walks you through searching for specific tables/views in preparation for including them in a replication task. You first search for tables that match specific criteria. Then you select the required tables/views from the search results to include them in the task. You can also carry out another search with new criteria and then add additional tables/views to the replication task.

After you finish searching, you can select tables/views for replication. Continue with Selecting Specific Tables/Views for Replication.

To search for tables/views to use in a replication task:

1. In **Designer** mode, click **Table Selection**.
2. In the **Select Tables** dialog box, if the source endpoint supports view selection, select one of the following:
   - **All** to search for tables and views
   - **Tables** to search for tables only
   - **Views** to search for views only

   Otherwise, skip to the next step.
3. From the **Schema** drop-down list, select a table/view schema.

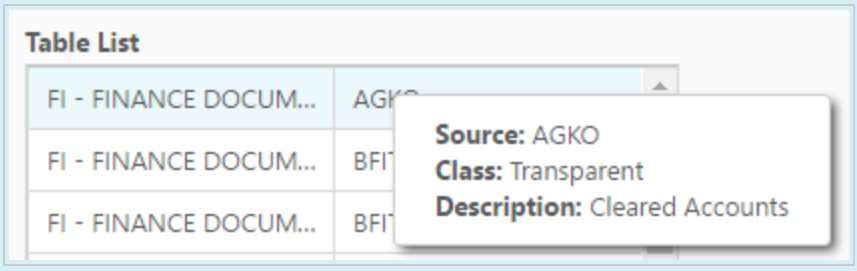
   Note When selecting tables from the SAP Application endpoint, "Business Groups" will appear instead of "Schema".

4. In the **Table/View** field, type the name or partial name of a table/view.

   Note You can also include special characters in your search string. For more information, see the Note in Creating a Record Selection Condition for One or More Columns.

5. Click **Search** to display a list of tables/views.

   Note When selecting tables from the SAP Application endpoint, the **Table Lis**t will display all of the tables in the selected Business Group. Hovering your mouse cursor over a table will display a tooltip as shown below.



The **Table List** field displays any table/view that matches the specified.

If the source endpoint supports view selection, an additional **Type** column indicates whether the database object is a table or a view.

6. Click **OK**.

Selecting Specific Tables/Views for Replication

This topic walks you through selecting tables/views to replicate in full. It assumes that you have already searched for the tables/views to use in the replication task. If you have not, start here: Searching for Tables/Views to use in a Replication Task

When you explicitly select tables/views, all selected tables/views are replicated in full unless you define transformations or filters for the table/view. If you need to make changes to the table/view structures in the target endpoint or if you only want to select specific columns, then you need to perform the procedures described in Defining Transformations for a Single Table/View and Using Filters respectively.

To explicitly select tables/views for replication:

1. In the **Select Tables** dialog box, from the **Table List** field, select one or more tables that you want to include in the replication task.
2. To select a table, click the button with a single right-facing arrowhead (**Add**).

   To select all tables in the **Table List**, click the button with two right-facing arrowheads **(Add All)**.

   The selected tables are added to the **Selected Tables** list.
3. Click **OK** to close the **Select Tables** or **Select Tables/Views** dialog box.
4. Click **Save** to make sure that Attunity Replicate saves the table information for this task.

Removing Specific Tables/Views from a Replication Task

This topic walks you through removing specific tables/views from the replication task.

To remove tables from the Selected Tables list:

1. From the **Selected Tables** list, select a table that you want to remove from the replication task and then click the button with a single left-facing arrowhead (**Remove**).
2. To remove all of the tables/views from the **Selected Tables** or **Selected Tables/Views** list, click the button with two left-facing arrowheads **(Remove All)**.
3. Click **OK** to close the **Select Tables** or **Select Tables/Views** dialog box.
4. Click **Save** to make sure that Attunity Replicate saves the table information for this task.

Creating Table/View Selection Patterns

This topic walks you through selecting tables/views using patterns. For example, you can include all tables/views that belong to the HR schema except for one or two tables/views that you exclude. You can also only exclude one or more table/view schemas or

tables/views. This replicates the entire endpoint, except for those tables/views that you excluded.

The following example shows a pattern that replicates all tables that are members of the HR schema except for the HR.EMPLOYEES table.

```
Include HR.%
Exclude HR.EMPLOYEES%
```

When you explicitly select tables/views, all selected tables/views are replicated in full unless you define transformations or filters for the table/view. If you need to make changes to the table/view structures in the target endpoint or if you only want to select specific columns, then you need to perform the procedures described in Defining Transformations for a Single Table/View and Using Filters respectively.

Note To view all of the tables/views included when you use a table selection pattern, click the **Full Table List** tab in Designer view. The **Full Table List** lists all of the tables/views included in any table pattern you defined as well as all explicitly selected tables/views. To view only patterns and explicitly selected tables/views, click the **Patterns and Selected Tables** tab in Designer view.

To create table/view selection patterns:

1. In the Designer view, in the **Select Tables/Views** dialog box, do any of the following:
   - Select a schema from the **Schema** drop-down list. All tables/views that belong to that schema are included in the table/view selection pattern.
   - Type the name or partial name of a table/view in the **Table/View** field. Any string that you enter here is included in the table/view selection pattern.
   - If the table/view that you type here is a member of the schema you selected in the **Schema** drop-down list, then you only have to type the name of the table/view.
   - If you did not select a schema or the table/view belongs to another schema, include the schema with the table name in the following format: `HR.Employees`, where `HR` is the schema.
2. Click **Include** to include all of the tables/views that match the selection criteria.
3. Click **Exclude** to exclude any tables that match the selection criteria.
4. Click **OK** to close the **Select Tables/Views** dialog box.
5. Click **Save** to make sure that Attunity Replicate saves the table/view information for this task.

Setting Load Order

You can set the load order for each of the selected tables. This may be useful, for example, if your selected tables list contains tables of different sizes and you want the smaller tables to be loaded before the larger tables. When a group of tables are set with the same load order, Replicate will load the tables according to the table ID.

Load order can be set and modified (see note below) in the following places:

- The **Select Tables** window (opened in Designer view by clicking the **Table Selection** button in the right of the console).
- The **Patterns and Select Tables** list in the right of the console (in Designer view).

Note Load order cannot be changed during a task. If you want to change the load order, first stop the task, then change the load order as desired, and finally reload the target.

Note Load order conflicts are handled as follows:

- If load order is set for a specific table and for a table pattern that includes that table, the load order priority for the specific table takes precedence.
- If several patterns include the same table and each of the patterns has a different load order priority, the pattern that Replicate handles first determines that table's load order priority.

To set the load order for a specific table or table selection pattern

1. Select the desired table in the **Selected Tables** list.

   -OR-

   Select the desired pattern (if defined) in the **Table Selection Patterns** list.
2. From the **Load Order** drop-down list, select one of the available priority levels (Lowest Priority, Low Priority, Normal Priority, High Priority, and Highest Priority).
3. This step is only relevant if you are setting load order in the **Select Tables** window. Click **OK** to save your settings and close the **Select Tables** window.

To set the same load order for multiple tables or table selection patterns

1. Select the desired tables in the **Selected Tables** list.

   -OR-

   Select the desired patterns (if defined) in the **Table Selection Patterns** list.
2. From any of the selected items' **Load Order** drop-down list, select one of the available priority levels.
3. This step is only relevant if you are setting load order in the **Select Tables** window. Click **OK** to save your settings and close the **Select Tables** window.

Editing a Replication Task

You can make changes to tasks that you previously created. Just open the task and make the changes in the same way that you did when you created the task.

To edit a task:

1. In **Tasks** view, select the task and click **Open**.

   The task opens, displaying the source and target endpoints and which tables have been selected for replication.
2. Continue with any of the following procedures:
   - Adding a Source and Target Endpoint to a Task
   - Adding Tables and/or Views to a Task
   - Defining Transformations for a Single Table/View
   - Using Filters
   - Task Settings

Searching for Tasks

In **Tasks** view, you can search for tasks by typing a sequence of letters in the **Filter Tasks** box above the tasks. For example, to search for all tasks with names that begin with "Oracle-to", type "or". Only tasks that match the search string are displayed.

Deleting a Replication Task

You can delete tasks that you created. To prevent complications, it is recommended that you not use the name of a deleted task for a new task you create. Such a task would be created with the same settings as the deleted task.

Note If you use a Microsoft SQL Server endpoint, a Microsoft SQL Server system administrator must delete the Microsoft SQL Server Replication Publisher definitions for the endpoint that was used in the task from SQL Server.

For more information, see the Limitations in the Microsoft SQL Server chapter.

To delete a task:

1. Stop the task that you want to delete.
2. In **Tasks** view, click **Delete**.

   The task is deleted.

Exporting and Importing Tasks

You can export replication tasks to a file. When exporting a task using the command line, all exported tasks are saved to the **imports** folder under <product_dir>/Attunity/Replicate/Data. When exporting a task using the Attunity Replicate Console, one of the following occurs (according to your browser settings):

- The task JSON file will be automatically downloaded to the default download location
- You will be prompted for a download location

You can import an export file (*.json) to another instance of the Attunity Replicate Server. This lets you use a task that you created in Attunity Replicate in a different environment. For example, if you created tasks in a development environment and now want to use the task in a production environment.

Importing and exporting a task can be done using either the command line or the Attunity Replicate Console. When exporting or importing a task using the command line, you must perform this task on the computer where Attunity Replicate is installed.

Note If you need to access the computer with Attunity Replicate from a remote computer, you can use a telnet connection.

When you export a task to a different environment, you may need to edit the task information. For example, you may need to change the connection string for an endpoint.

- Exporting Tasks
- Editing an Exported (json) File

Exporting Tasks

The following section explains how to export a task using either the Attunity Replicate Console or the command line.

To export a task using the Attunity Replicate Console:

1. Switch to **Tasks** view (make sure you're in **Designer** mode).
2. Do one of the following:
   - In **TASKS** tab, select the task you want to export and then either click the **Export** toolbar button or right-click the task and select **Export**.

     -OR-
   - In the ***TASK_NAME*** tab (opened when a task is being edited), click the **Export Task** toolbar button.

   Depending on your browser settings, one of the following will occur:
   - The task JSON file will be automatically downloaded to the default download location
   - You will be prompted for a download location. In this case, save the JSON file to your preferred location.

To export a task using the command line:

1. From the Attunity Replicate computer where you defined the task you want to import, open the Attunity Replicate command line console by doing the following:

   On Windows: Go to **All Programs** in the **Start** menu and point to **Attunity Replicate**, then to **Utilities** and then select **Attunity Replicate Command Line**.

A command-line console is displayed with the correct prompt for Attunity Replicate.

Note: You can also open the Windows command-line console and change the directory to the following:

`<Full path to the Attunity Replicate root folder>\bin`

For example, to use the path to the folder or directory where Attunity Replicate is installed by default, type: `C:\Program Files\Attunity\Replicate\bin`.

On Linux: Enter the Linux computer, then type the following before you continue:

```
source ./arep_login.sh
```

2. At the prompt in the command-line console, type the following:

   ```
   repctl exportrepository task=task_name [folder_name=path]
   ```

   By default, a file called `<task_name>.json` containing the exported task settings is created in the `<product_dir>\data\imports` folder. If you want the file to be created in a different folder, include the `folder_name=path` parameter in the command.

   After the file has been created, you can import it into another Attunity Replicate instance as described in Importing Tasks.

   Note If the `<product_dir>\data` folder was installed in a non-default location during the installation - OR - if it was later moved to a non-default location, you need to tell Replicate where the folder is located.

   This is done by including the `-d <data_folder>` parameter in the command.

   Example:

   ```
   repctl -d D:\Data exportrepository task=mytask
   ```

Importing Tasks

The following section explains how to export a task using either the Attunity Replicate Console or the command line.

Note Importing a Hadoop target from earlier Replicate versions to Replicate 5.1.1 changes the Hive Access mode to ODBC.

If you were previously using WebHCat to access Hive, you can change it back to WebHCat after the import completes.

See also Using Hadoop as a Source.

To import a task using the Attunity Replicate Console:

1. Switch to **Tasks** view (make sure you're in **Designer** mode).
2. Click the **Import Task** toolbar button.

   The **Import Task** dialog box opens.
3. Browse to the task JSON file you want to import and then click **Import Task**.

   The task is imported.

To import a task using the command line:

1. From the Attunity Replicate computer where you want to export the task, open the Attunity Replicate command line console by doing the following:

   Go to **All Programs** in the **Start** menu and point to **Attunity Replicate**, then to **Utilities** and then select **Attunity ReplicateCommand Line**.

   A command-line console is displayed with the correct prompt for Attunity Replicate.

   Note You can also open the Windows command-line console and change the directory to the following:

   `<product_dir>\Attunity Replicate>\bin`

   For example to use the path to the directory where Attunity Replicate is installed by default, type: `C:\Program Files\Attunity\Replicate\bin`.

2. At the prompt in the command-line console, type the following:

   `repctl importrepository json_file=<Full path to the exported *.json file>`

   Example:

   `repctl importrepository json_file=C:\Temp\many_tables`

   Important: The name of the JSON file must be specified *without* the .JSON extension.

   The export utility automatically appends the extension .JSON to exported files, thus eliminating the need to add the extension manually.

   The exported *.json file will be located in the `<data_directory>\imports` folder or directory on the original computer where the task was exported or in the folder specified by the `folder_name` parameter in the export command.

   Example:

   `<product_dir>\data\imports\many_tables.json`

   Note If the `<product_dir>\data` folder was installed in a non-default location during the installation - OR - if it was later moved to a non-default location, you need to tell Replicate where the folder is located.

   This is done by including the `-d <data_folder>` parameter in the command.

   Example:

   `repctl -d D:\MyData importrepository json_file=C:\mytask.json`

   If you are importing this task into a different environment, you should copy the file to a location on the second Attunity Replicate computer and then import the file from there.

   In many cases, when you import the task into a different environment, you will need to make changes to the task. For example, you may need to change the connect strings for

the endpoints in the task or change the user password. In that case, you will need to edit the *.json file.

See Editing an Exported (json) File for more information.

Editing an Exported (json) File

You can open the *.json file in any plain text editor. It is possible to make changes to any of the sections in the file; however, be sure that you only change the data and not the field names. For example, the entry `"name"::DB_Name"` displays the name field for a source table in a defined endpoint. In this case, you can change the data "DB_Name" but not the included metadata (`"name"`).

Important: Make any changes to the *.json file *before* you carry out the import operation.

Note Information about the endpoints, tables, tasks, task settings, and logger settings should be changed using the Attunity Replicate Console after the file is imported.

To be able to use the new task, you will need to make changes to the endpoint password and connection strings by editing the *.json file. See Making Changes to the Endpoint Connection Information for more information.

Making Changes to the Endpoint Connection Information

In the `"endpoints"` section, you can make changes to the connection string and the password. The following is an example of the `"endpoints"` section of the *.json file.

```
"endpoints": [{
"name": "Oracle_Source",
"type": "Oracle",
"connection_string": "server= bee01-xp:1521/xe;username=SYSTEM",
"authenticator": "
{01000000D08C9DDF0115D1118C7A00C04FC297EB010000003EA495B32CAAE14CB9777B96B3
CC00B300000000020000000000003660000A800000010000000002765A3287AB56447DA31508F7
1CE627000000000004800000A00000001000000088D5C1BBD615BEEEAF5FAC1B9B0E20800800
000075D89177A9C6F11B1400000047B3110B80920DD9EB0A5FABA05679979B78DDD0}",
"role": "SOURCE"
}, {
"name": "SQLSERVER_Target",
"type": "SQLServer",
"connection_string": "server=bee01-
xp;endpoint=tempdb;WindowsAuthentication=Y;CDCBCP=Y;FullloadBCP=Y;BCPPacket
Size=16384",
"role": "TARGET"
```

To change the connect string:

1. In the *.json file, under `"endpoints"`, find `"connection string"`.

   For example, `"connection_string": "server= bee01:1521/xe;username=SYSTEM"`.
2. Change the information after the equal signs (=) as necessary.

   For example, if the endpoint in the new environment is on a computer called B2, change `server=bee01` to `server=B2`.

   Important: Make sure that the connect string remains between the quotation marks (").

To change an endpoint password:

1. In the *.json file, under `"endpoints"`, find `"authenticator"`.

   For example, `"authenticator": "{01000000D08C9DDF0115D1118C7A00C04FC297EB010000003EA495B32CAAE14CB9777B96B3CC00B3000000000200000000000003660000A8000000100000002765A3287AB56447DA31508F71CE627000000000004800000A00000001000000088D5C1BBD615BEEEAF5FAC1B9B0E20800800000075D89177A9C6F11B1400000047B3110B80920DD9EB0A5FABA05679979B78DDD0}"`.

   Note The password is presented as an encrypted string.
2. Change the password string to the relevant password for the endpoint in the new environment. Type the new password using plain text exactly as it is configured in the endpoint. For example, `8yTkLMt`.

   When you save the file and then import it to the new environment, the password is encrypted automatically.

8 | Adding and Managing Source Endpoints

This chapter describes how to configure source endpoint settings.

In this chapter:

Using Oracle as a Source

This section describes how to set up and use an Oracle database as a source in a replication task.

In this section:

Supported Oracle Database Editions

Before you begin to work with an Oracle database as a source or target in Attunity Replicate, make sure that the Oracle database with the tables that are necessary for replication is available in your system. Attunity Replicate supports the following Oracle database editions:

- Oracle Enterprise Edition
- Oracle Standard Edition
- Oracle Express Edition
- Oracle Personal Edition

Prerequisites

Before you can work with an Oracle endpoint, make sure the prerequisites listed in this section have been met.

- On Windows systems, install Oracle Instant Client for Microsoft Windows (x64) Version 11.2.0.3.0 and above.

  Note Support for the XMLTYPE data type requires the full Oracle Client.

- On Linux systems, install Oracle Instant Client for Linux (x86-64) Version 11.2.0.3.0 and above.

  Note Support for the XMLTYPE data type requires the full Oracle Client.

  In addition, if not already included in your system, you need to create a symbolic link in the `$Oracle_Home\lib` directory. This link should be called `libclntsh.so`, and should point to a specific version of this file. For example, on an Oracle 12c client:

  ```
  lrwxrwxrwx 1 oracle oracle 63 Oct 2 14:16 libclntsh.so ->
  /u01/app/oracle/home/lib/libclntsh.so.12.1
  ```

- Additionally, the `LD_LIBRARY_PATH` environment variable should be appended with the Oracle `lib` directory and added to the `site_arep_login.sh` script.

Limitations

The following limitations apply:

- Function-based indexes are not supported.
- If you are managing supplemental logging and you carry out transformations on any of the columns, you must be sure that supplemental logging is activated for all fields and columns.
- Connecting to a CDB is not supported.
- The AR_H_USER header column is supported only for Oracle database version 11.2.0.3 and higher. In Oracle database version 10, the value for this column may not be correct. For information on using header columns, see Header Columns.
- The rename table <*table name*> to <*new table name*> syntax is supported by Attunity Replicate when using Oracle version 11 and higher.
- Data changes resulting from partition/sub-partition operations (ADD, DROP, EXCHANGE and TRUNCATE) will not be replicated. To replicate such changes, you need to reload the table. Any future data changes to newly added partitions will be replicated to the target *without* needing to reload the table again. However, operations to UPDATE old data records (in these partitions) will generate a "0 rows affected" warning.
- Data changes resulting from the "CREATE TABLE AS..." statement is not supported. However, the new table *will* be created on the target.
- When Limited-size LOB mode is enabled, empty LOBs on the Oracle source are replicated as NULL values. For more information on Limited-size LOB mode, see Task Settings|Metadata.
- Changes made by the Oracle DBMS_REDEFINITION package - e.g. table metadata and the OBJECT_ID) - will not be captured by Attunity Replicate.
- To enable change processing from an Oracle standby database, the database must be a physical standby database with Active Data Guard enabled (available from Oracle 11g and above).
- Empty BLOB/CLOB columns are mapped to NULL on the target.

- During Change Processing, columns without supplemental logging that are not updated will be inserted as NULL in the Change Table.
- During Change Processing, batch updates to numeric columns defined as a Primary Key are not supported.

  Example of an unsupported UPDATE command:

  ```
  UPDATE tableX set ID=ID+1;
  ```

  Where `tableX` is the table name and `ID` is a numeric column defined as a Primary Key.
- Data in LONG and LONG RAW column cannot exceed 64k. Any data that is larger than 64k will be truncated.
- Tables whose names contain apostrophes cannot be replicated.
- Change Data Capture (CDC) is not supported from dynamic views.
- When running a Full Load and Appy Changes task on a physical standby Oracle database, Replicate may not be able to properly synchronize the Full Load and Appy Changes processes, which could result in missing operations.
- When using LogMiner to access the redo logs, the following limitations apply:
  - The UPDATE statement is not supported for XMLTYPE and LOB columns.
  - The DDL statement `ALTER TABLE ADD <column> <data_type> DEFAULT <>` does not replicate the default value to the target and the new column is set to NULL. Note that this may happen even if the DDL that added the column was executed in the past. If the new column is nullable, Oracle updates all the table rows before logging the DDL itself. As a result, Attunity Replicate captures the changes but does not update the target. As the new column is set to NULL, if the target table has no Primary Key/Unique Index, subsequent updates will generate a "zero rows affected" message.
  - SHRINK SPACE operations are not supported
  - Connecting to a PDB using LogMiner is not supported. Therefore, if you want to connect to a PDB, make sure that the **Access redo logs via Binary reader** option is selected in the **Advanced** tab.
- When using Binary Reader to access the redo logs, the following limitations apply:
  - Table clusters are not supported.
  - Only table-level SHRINK SPACE operations are supported. These include the full table, partitions, and sub-partitions.
  - Index-organized tables with an overflow segment are not supported in CDC mode.
  - Implementing online redo logs on raw devices is not supported.
  - When the Oracle database is installed on Linux, it is strongly recommended *not* to use a file system that is mounted for 'directio'.
  - Changes to Index-organized tables with key compression are not supported.

Required Permissions

To use an Oracle source in an Attunity Replicate task, the user specified in the Attunity Replicate Oracle endpoint connection settings must be granted the following privileges in

the Oracle database:

Note If any of the required privileges cannot be granted to a V$xxx, then grant them to the V_$xxx.

- SELECT ANY TRANSACTION
- SELECT on V_$ARCHIVED_LOG
- SELECT on V_$LOG
- SELECT on V_$LOGFILE
- SELECT on V_$DATABASE
- SELECT on V_$THREAD
- SELECT on V_$PARAMETER
- SELECT on V_$NLS_PARAMETERS
- SELECT on V_$TIMEZONE_NAMES
- SELECT on V_$TRANSACTION
- SELECT on ALL_INDEXES
- SELECT on ALL_OBJECTS
- SELECT on DBA_OBJECTS - Required if the Oracle version is earlier than 11.2.0.3.
- SELECT on ALL_TABLES
- SELECT on ALL_USERS
- SELECT on ALL_CATALOG
- SELECT on ALL_CONSTRAINTS
- SELECT on ALL_CONS_COLUMNS
- SELECT on ALL_TAB_COLS
- SELECT on ALL_IND_COLUMNS
- SELECT on ALL_LOG_GROUPS
- SELECT on SYS.DBA_REGISTRY
- SELECT on SYS.OBJ$
- SELECT on SYS.ENC$
- SELECT on DBA_TABLESPACES
- SELECT on ALL_TAB_PARTITIONS
- SELECT on ALL_ENCRYPTED_COLUMNS
- If views are exposed: SELECT on ALL_VIEWS

Grant the following additional privilege (for each replicated table) when you are using a specific table list:

- SELECT on <any-replicated-table>;

Grant the following additional privilege when using a pattern for the table list:

- SELECT ANY TABLE;

Grant the following additional privilege (for each replicated table) when Attunity Replicate adds supplemental logging automatically (the default behavior) and you are using a specific table list. For information on how to turn off supplemental logging, see Setting Advanced Connection Properties.

- ALTER on <any-replicated-table>;

Grant the following additional privilege when Attunity Replicate adds supplemental logging automatically (the default behavior). For information on how to turn off supplemental logging, see Setting Advanced Connection Properties Using LogMiner.

- ALTER ANY TABLE;

Access Privileges when using LogMiner to Access the Redo Logs

If you are using LogMiner to access the Redo logs, grant the following privileges.

- CREATE SESSION
- EXECUTE on DBMS_LOGMNR
- SELECT on V_$LOGMNR_LOGS
- SELECT on V_$LOGMNR_CONTENTS
- LOGMINING

Note This privilege is only required for Oracle 12c and above.

Access Privileges when using Binary Reader to Access the Redo Logs

The following privileges should be granted when using Binary Reader to access the Redo logs:

- CREATE SESSION
- SELECT on v_$transportable_platform

  Grant the SELECT on v_$transportable_platform privilege if the Redo logs are stored in ASM and accessed by Replicate from ASM.
- CREATE ANY DIRECTORY

  Attunity Replicate uses following Oracle file access features:
- BFILE read - Used when Replicate does not have file-level access to the Redo logs, and the Redo logs are not accessed from ASM.
- DBMS_FILE_TRANSFER package - Used to copy the Redo log files to a temporary folder (in which case the EXECUTE ON DBMS_FILE_TRANSFER privilege needs to be granted as well)
- DBMS_FILE_GROUP package - Used to delete the Redo log files from a temporary/alternate folder (in which case the EXECUTE ON DBMS_FILE_GROUP privilege needs to be granted as well).

Oracle file features work together with Oracle directories. Each Oracle directory object includes the name of the folder containing the files which need to be processed.

If you want Replicate to create and manage the Oracle directories, you need to grant the CREATE ANY DIRECTORY privilege specified above. Note that the directory names will be prefixed with attu_. If you do not grant this privilege, you need to create the corresponding directories manually. In you create the directories manually and the Oracle user specified in the Oracle Source endpoint is not the user that created the Oracle Directories, grant the READ on DIRECTORY privilege as well.

If the Oracle source endpoint is configured to copy the Redo log files to a temporary folder, and the Oracle user specified in the Oracle source endpoint is not the user that created the Oracle directories, the following additional privileges are required:

- READ on the Oracle directory object specified as the source directory
- WRITE on the directory object specified as the destination directory in the copy process

See also: Setting Advanced Connection Properties.

Supported Encryption Methods

The table below lists which encryption methods Attunity Replicate supports when working with an Oracle source database.

Table 8.1 | Supported Oracle Encryption Methods

Redo Logs Access Method	TDE Tablespace	TDE Column
Binary Reader	Yes	Yes
LogMiner	Yes	Yes

Supported Compression Methods

The table below lists which compression methods Attunity Replicate supports when working with an Oracle source database. As the table shows, compression support depends both on your Oracle database version and whether or not Attunity Replicate is configured to use LogMiner to access the redo logs.

Table 8.2 | Supported Oracle Compression Methods

Version	Basic	OLTP	HCC (from Oracle 11g R2)	Others
Oracle 10	No	n/a		No
Oracle 11 and above - Binary Reader	Yes	Yes	Yes See Note below.	No
Oracle 11 and above - LogMiner	Yes	Yes	Yes	*Yes

*Any compression method supported by LogMiner

Note When the Oracle source endpoint is configured to use Binary Reader, the Query Low level of the HCC compression method is only supported in the **Full Load** task mode.

Redo Log Files - Access Method Guidelines

The Replicate Oracle source endpoint can be configured to access online and archived Oracle redo log files using either LogMiner (Oracle's built-in tool) or Binary Reader (Replicate's high-speed redo logs reader).

Generally, it is recommended to use Binary Reader as it is more efficient, faster, and uses less resources.

Binary Reader is especially recommended in the following situations:

- The volume of changes in the redo log is more than 30GB/hour
- The volume of changes is between 10GB/hour and 30GB/hour and the changes need to be processed (i.e. replicated to the target) as fast as possible.
- There are multiple tasks replicating from the same source. Using LogMiner is less efficient in this case, as it accesses the redo logs via the database, thereby consuming additional database resources.

Both Binary Reader and LogMiner are subject to certain limitations, support different compression methods, and require different access permissions. It is therefore strongly recommended to review the relevant sections before configuring the endpoint settings. In the end, your decision whether to use Binary Reader or LogMiner may be based on a limitation that exists in one but not the other, the way your data is compressed, or the permissions that you are willing to grant the Replicate user.

Handling Shrink Space Operations

When a SHRINK SPACE operation occurs, Replicate will capture all of the changes logged to the redo log as a result of the operation and ignore them.

The following message will appear in the task's log file:

```
Operations generated by the SHRINK SPACE process were ignored.
```

Monitoring Considerations:

When Replicate captures changes resulting from a SHRINK SPACE operation, the task's Incoming Changes bar will indicate an unusually large number of changes. However, these changes will not be reflected in the Applied Changes pie chart or the Applied Changes Details table.

See also Limitations for SHRINK SPACE operations.

Oracle Source Data Types

The Oracle database for Attunity Replicate supports most Oracle data types. The following table shows the Oracle source data types that are supported when using Attunity Replicate

and the default mapping to Attunity Replicate data types.

For information on how to view the data type that is mapped in the target, see the section for the target database you are using.

For additional information about Attunity Replicate data types, see Replicate Data Types.

Table 8.3 | Supported Oracle Data Types with Mapping to Attunity Replicate Data Types

Oracle Data Types	Attunity Replicate Data Types
BINARY_FLOAT	**REAL4**
BINARY_DOUBLE	**REAL8**
BINARY	**BYTES**
FLOAT (P)	Precision < or = 24: **REAL4** Precision > 24: **REAL8**
NUMBER (P,S)	When scale is < 0: **REAL8**
NUMBER according to the "Expose number as" property in the Attunity Replicate Oracle source database settings.	When scale is 0 and: Precision = 0: **REAL8** Precision < or = 2: **INT1** Precision >2 and <or = 4: **INT2** Precision >4 and <or = 9: **INT4** Precision > 9: **NUMERIC** If precision > or = scale: **NUMERIC** In all other cases: **REAL8**
DATE	**DATETIME**
INTERVAL_YEAR TO MONTH	**STRING (with interval year_to_month indication)**
INTERVAL_DAY TO SECOND	**STRING (with interval day_to_second indication)**
TIME	**DATETIME**
TIMESTAMP	**DATETIME**
TIMESTAMP WITH TIME ZONE	**STRING (with timestamp_with_timezone indication)**
TIMESTAMP WITH LOCAL TIME ZONE	**STRING (with timestamp_with_local_timezone indic-**

Table 8.3 | Supported Oracle Data Types with Mapping to Attunity Replicate Data Types (Cont.)

Oracle Data Types	Attunity Replicate Data Types
	ation)
CHAR	**STRING**
VARCHAR2	**STRING**
NCHAR	**WSTRING**
NVARCHAR2	**WSTRING**
RAW	**BYTES**
REAL	**REAL8**
BLOB To use this data type with Attunity Replicate, you must enable the use of BLOBs for a specific task. BLOB data types are supported only in tables that include a primary key. For more information, see **LOB support** in **Task Settings**/Metadata.	**BLOB**
CLOB To use this data type with Attunity Replicate, you must enable the use of CLOBs for a specific task. During CDC, CLOB data types are supported only in tables that include a primary key. For more information, see **LOB support** in **Task Settings**/Metadata.	**CLOB**
NCLOB To use this data type with Attunity Replicate, you must enable the use of NCLOBs for a specific task. During CDC, NCLOB data types are supported only in tables that include a primary key. For more information, see **LOB support** in **Task Settings**/Metadata.	**NCLOB**
LONG The LONG data type is not supported in Batch Optimized Apply mode. To use this data type with Attunity Replicate, you must enable the use of LOBs for a specific task.	**CLOB**

Table 8.3 | Supported Oracle Data Types with Mapping to Attunity Replicate Data Types (Cont.)

Oracle Data Types	Attunity Replicate Data Types
During CDC, LOB data types are supported only in tables that have a primary key. For more information, see **LOB support** in **Task Settings**/Metadata.	
LONG RAW The LONG RAW data type is not supported in Batch Optimized Apply mode. To use this data type with Attunity Replicate, you must enable the use of LOBs for a specific task. During CDC, LOB data types are supported only in tables that have a primary key. For more information, see **LOB support** in **Task Settings**/Metadata.	**BLOB**
XMLTYPE **Note:** Support for the XMLTYPE data type requires the full Oracle Client (as opposed to the Oracle Instance Client). When the target column is a CLOB, both full lob mode and limited lob mode are supported (depending on the target). For more information, see **LOB support** in **Task Settings**/Metadata.	**CLOB**

Non-Supported Data Types

Source Oracle tables with columns of the following Oracle data types cannot be replicated. Replicated columns with these data types will show as null:

- BFILE
- ROWID
- REF
- UROWID
- ANYDATA
- SDO_GEOMETRY
- Nested Table
- User-defined data types

Note

- Virtual columns are not supported.
- As the ROWID data type is not supported, materialized views based on a ROWID column are also not supported.

Homogeneous Replication

With the exception of the LONG and LONG RAW data types, when replicating from an Oracle source to an Oracle target, all of the source and target data types will be identical. The LONG data type will be mapped to CLOB and the LONG RAW data type will be mapped to BLOB. It should be noted that, as of Oracle 9.0, the LONG and LONG RAW data types are no longer supported by Oracle.

Additionally, Primary/Unique Index names are preserved during homogeneous replication.

Note In homogeneous replication, the source data first passes through the Attunity Replicate data type and is therefore subject to any limitations of that type.

For information on Replicate data types and their limitations (where relevant), see Replicate Data Types.

For information on which Replicate data types the source data passes through when replicating from Oracle, see the Oracle to Attunity Replicate data types mapping table described earlier.

Preparing the Oracle Database for Replication

The following topics describe the configuration requirements for using an Oracle database with Attunity Replicate as a source. An Oracle DBA should know how to carry out these tasks.

- Provide Oracle Account Access
- Ensure that ARCHIVELOG Mode is On
- Setting up Supplemental Logging

Provide Oracle Account Access

You must provide Oracle account access to the Attunity Replicate user. This user must have read/write privileges on the Oracle database. For information on setting up access to the Oracle account, see Required Permissions.

Ensure that ARCHIVELOG Mode is On

Oracle can be run in two different modes: the `ARCHIVELOG` mode and the `NOARCHIVELOG` mode. To use the Oracle logs with Attunity Replicate, run the database in `ARCHIVELOG` mode. If the log is not set to `ARCHIVELOG` mode, then execute the following query:

```
ALTER database ARCHIVELOG
```

Note that if your Oracle database instance is on Amazon RDS, a different command needs to be executed. For more information, see Working with Oracle on Amazon RDS and Working with Oracle on Amazon RDS in Working with Oracle on Amazon RDS.

Setting up Supplemental Logging

Supplemental logging must be enabled for the Oracle database.

Note You can automatically set up supplemental logging in the **Advanced** tab of the Oracle database dialog box. If you select this option, you do not have to carry out the following procedure. For more information, see Setting Advanced Connection Properties Using LogMiner.

Set up supplemental logging as described in the steps below.

Step 1: Check that supplemental logging is enabled for the database

1. Run the following query:

   ```
   SELECT name, value, description FROM v$parameter WHERE name =
   'compatible';
   ```

   The returned result should be from `GE to n.n.n` where *n.n.n* is the Oracle database version (e.g. 10.0.0).

   Note For Replicate to work, the parameter value must match the real version of the database.

2. Run the following query:

   ```
   SELECT supplemental_log_data_min FROM v$database;
   ```

   The returned result should be `YES` or `IMPLICIT`.

   Enable supplemental logging by executing the following query:

   ```
   ALTER DATABASE ADD SUPPLEMENTAL LOG DATA
   ```

   Note If your Oracle database instance is on Amazon RDS, a different command needs to be executed. For more information, see Working with Oracle on Amazon RDS.

Step 2: Make sure that the required supplemental logging is added for each table

1. If a Primary Key exists, supplemental logging must be added for the Primary Key either by using the format to add supplemental logging on the Primary Key, or by adding supplemental logging on the Primary Key columns.
2. If no Primary Key exists and the table has a single Unique Index, then all of the Unique Index's columns must be added to the supplemental log. Using `SUPPLEMENTAL LOG DATA (UNIQUE INDEX) COLUMNS` does not add the Unique Index columns to the log.

3. If no Primary Key exists and the table has multiple Unique Indexes, Attunity Replicate will select the first Unique Index. Attunity Replicate will use the first index in an alphabetically ordered ascending list. Supplemental logging must be added on the selected index's columns. Using `SUPPLEMENTAL LOG DATA (UNIQUE INDEX) COLUMNS` does not add the Unique Index columns to the log.
4. If there is no Primary Key and no Unique Index, supplemental logging must be added on all columns.

   Note When the target table Primary Key/Unique Index is different than the source table Primary Key/Unique Index, the user needs to add supplemental logging manually on the source table columns that comprise the target table Primary Key/Unique Index.

5. If you change the target table primary key, the supplemental logging must be added on the selected index's columns instead of the columns of the original primary key/unique index.

Step 3: If a filter or transformation is defined for the table, additional logging might be necessary

Note If ALL COLUMNS supplemental logging has been added to the table, there is no need to add any additional logging.

If the table has a Unique Index or a Primary Key, you also need to add supplemental logging on each column that is involved in a filter or transformation (if those columns are different than the Primary Key or Unique Index columns).

Note If a transformation uses only one column, this column may not be added to a supplemental logging group. For example, "A+B" needs both columns to be added, whereas substring(A, 10) does not need "A" to be added.

One method of setting up both Primary Key/Unique Index supplemental logging and supplemental logging on specific columns is to add `USER_LOG_GROUP` supplemental logging only on the Primary Key/Unique Index columns and on the columns that are filtered or transformed.

For example, to replicate a table named `EXAMPLE.TABLE` with Primary Key ID and filter by column `NAME`, you can run a command similar to the one below to create the log group supplemental logging:

```
ALTER TABLE EXAMPLE.TABLE ADD SUPPLEMENTAL LOG GROUP example_log_group
(ID,NAME) ALWAYS;
```

Step 4: When the Insert the missing target record **Apply Conflicts option is selected, supplemental logging must be enabled for ALL the source table columns.**

Working with Oracle on Amazon RDS

Attunity Replicate supports Oracle 11.2.0.2.v7 and above on Amazon RDS as a source database and in LogMiner mode only. This section details the requirements for working with Oracle on Amazon RDS.

Setting Up Supplemental Logging

Attunity Replicate requires database-level supplemental logging to be enabled. To enable database-level supplemental logging, execute the following command:

```
exec rdsadmin.rdsadmin_util.alter_supplemental_logging('ADD');
```

Although not required, examples of additional commands that you can execute to change the supplemental logging attributes include:

```
exec rdsadmin.rdsadmin_util.alter_supplemental_logging('ADD','ALL');
```

```
exec rdsadmin.rdsadmin_util.alter_supplemental_logging('DROP','PRIMARY KEY');
```

Enabling Automatic Backups

In **Step 5: Management Options** of setting up your Oracle database instance, set the **Enabled Automatic Backups** option to **Yes**.

Setting Up Archiving

To retain archived redo logs of your Oracle database instance (which will allow Attunity Replicate to retrieve the log information using LogMiner), execute the following command (example 24 hours):

```
exec rdsadmin.rdsadmin_util.set_configuration('archivelog retention hours',24);
```

Make sure that your storage has sufficient space for the archived redo logs during the specified period.

Setting General Connection Properties

This section describes how to configure general connection properties. For an explanation of how to configure advanced connection properties, see **Setting Advanced Connection Properties** below.

Note

» Oracle can also be used as a target database. For information on using Oracle as a target, see Setting General Connection Properties.

» You can also use Oracle files as a source or target. For more information, see Using a File as a Source.

To add an Oracle source endpoint to Attunity Replicate

1. In **Tasks** view, click **Manage Endpoint Connections** to open the **Manage Endpoints Connections** dialog box. Then click the **New Endpoint Connection** button.
2. In the **Name** field, type a name for your database. This can be any name that will help to identify the database being used.
3. In the **Description** field, type a description that helps to identify the Oracle database. This is optional.
4. Select **SOURCE** as the database **role**.
5. Select **Oracle** as the database **Type**.
6. Type the Oracle **Connection String** for the Oracle database you want to work with. You can type the connect string in any Oracle format, for example:

   `//host:port/service name`

   Where:

   - `host`: This is the name or IP address for the computer with the Oracle database that you are using. For example, `johnboy_W7` or `255.255.255.0`.
   - `port`: (optional) This is the TNS Listener Port number for the computer with the Oracle database that you are using. If you do not enter a port number the default Oracle TNS Listener port is used.
   - `service name`: (optional) This is the service name for the computer with the Oracle database you are using. If you do not enter a service name the default service name is used.

   You can also enter an Oracle Net keyword-value pair. For example:

   "(DESCRIPTION=(ADDRESS=(PROTOCOL=tcp) (HOST=dlsun242) (PORT=5521))
   (CONNECT_DATA=(SERVICE_NAME=bjava21)))"

   Note When working with a Multitenant environment, the connection string should specify a specific PDB.

   Limitations:

   - Connecting to the CDB is not supported.
   - Oracle does not support using PDB with LogMiner. Therefore, if you want to connect to a PDB, make sure that the **Use LogMiner to access redo logs** option is disabled in the **Advanced** tab.

   Specifying Separate Connection Strings for Different RAC Instances

   If the Oracle endpoint is configured to use Binary Reader and the node to which Replicate is connected cannot access the logs created by the other cluster nodes, you need to specify a separate connection string for each RAC instance.

When the redo logs are stored in ASM, the connection string syntax is as follows:

```
[<common ASM connection string>,] <thread id> <thread ASM connection string>, <thread id> <thread ASM connection string>...
```

Note If no `<common ASM connection string>` is specified, all the RAC instances should be defined in the ASM connection.

When using Oracle Binary Reader to access the redo logs, the connection string syntax is as follows:

```
<Oracle connection string>[, <thread id> <thread BFILE connection string>, <thread id> <thread BFILE connection string> ...]
```

`<Oracle connection string>` is mandatory. If specified, the `<thread BFILE connection string>` will be used instead of the `<Oracle connection string>`.

7. Type the Oracle authentication information (User Name, Password) for the authorized user for this Oracle database. If you do not know this information, see your Oracle database Administrator (DBA).

Note This information is case sensitive.

Important: Make sure that the Oracle user entered in the Oracle Authentication section has the correct access privileges. For information on how to provide the required privileges, see Required Permissions.

Configuring Replicate to Automatically Replace the User-Entered Password

To prevent illicit database activity by unauthorized third-parties, Replicate can be configured to automatically replace the user-entered password with a strong random password.

Note Clicking the "Test Connection" button will verify the connection using the original password. The password will be automatically changed the first time the task runs.

To utilize this feature, the password must be defined both in the Replicate endpoint settings *and* on the actual database, in the following format:

```
replace:your_password
```

Example:

```
replace:k$RJdg7!S&ib
```

Defining Multiple Endpoints to use the same Automatically Changed Password

In Attunity Replicate, more than one endpoint may be configured to connect to the same database server.

To allow multiple endpoints to use the same (automatically changed) credentials, the password in one of the endpoints needs to defined. Then, each of the other endpoint connections needs to be configured to reference that endpoint.

Note A source endpoint cannot reference a target endpoint, only another source endpoint. Likewise, a target endpoint cannot reference a source endpoint, only another target endpoint.

Additionally, an endpoint cannot reference another endpoint that uses a different database server.

To configure an endpoint to use the automatically changed credentials of another endpoint:

a. In the **User name** field, enter the user name in the following format:

   `ref:endpoint_name`

   Where `endpoint_name` is the name of the endpoint connection whose password was automatically changed.

b. In the **Password** field, specify the password *before* it was automatically changed and without the "`replace`" prefix.

   Example:

   If the original password is:

   `replace:54lakrfgnier3!`

   Specify:

   `54lakrfgnier3!`

Setting Advanced Connection Properties

The Replicate Oracle source endpoint can be configured to access online and archived Oracle redo log files using either LogMiner (Oracle's built-in tool) or Binary Reader (Replicate's high-speed redo logs reader).

The following sections describe how to do this:

- Setting Advanced Connection Properties Using LogMiner
- Setting Advanced Connection Properties Using Binary Reader

Note For guidelines on choosing which redo logs access method to use, see Redo Log Files - Access Method Guidelines.

Setting Advanced Connection Properties Using LogMiner

This section describes which properties are available in the **Advanced** tab when using LogMiner to access the redo logs. For information on which properties are available in the **Advanced** tab when using LogMiner to access the redo logs, see Setting Advanced Connection Properties Using Binary Reader.

Note If your Oracle database version precedes 10.2.0.4 (i.e. version 10.1.x to 10.2.0.3), you must use LogMiner to access the redo logs.

- **Automatically add supplemental logging**: Select this to automatically set up supplemental logging for the Oracle database. This option is available when Binary Reader is selected as the redo logs access method.

  For more information on supplemental logging, see Setting up Supplemental Logging.
- Under the **Access redo logs via** label, choose **LogMiner**. Changes will be captured using the LogMiner utility (the default).
- **Secret Store encryption entries:** When some of the columns in the tables that you intend to replicate are encrypted you need to specify the Oracle Wallet encryption keys and their values. **See** Finding the Wallet Entry used for TDE Column Encryption in a Specific Table **.**
- **Retry interval**: Use the counter or type the number of seconds that the system waits before resending a query.
- **Archived redo logs destination identifier**: The destination of the archived redo logs. The value should be the same as the `DEST_ID` number in the `V$archived_log` table.

  Note When working with multiple log destinations (`DEST_ID`), you should specify an **Archived redo logs location identifier** that represents archived logs that can be accessed by Replicate. If the **Archived redo logs location identifier** is not specified, Replicate will use the minimal existing `DEST_ID`.

- **Expose NUMBER as**: Select a precision-scale combination, FLOAT or VARCHAR. Attunity Replicate supports any precision-scale combination supported by Oracle. By default, the NUMBER data type is converted to precision 38, scale 10.

  Note If precision is 39 or above, select VARCHAR.

  Note The "Expose NUMBER" definition in the Oracle database is used for the NUMBER data type only (without the explicit precision and scale definition).

- **Use archived redo logs only**: When this option is selected, Attunity Replicate will only access the archived redo logs. If the archived redo logs ares stored on ASM only,

the Attunity Replicate user needs to be granted the ASM privileges described in Setting Advanced Connection Properties Using LogMiner.

- **Replicate tablespace names:** Select this option to replicate the source tablespace names to an Oracle target. When this option is selected, the **Table tablespace** name will also be shown in the **General** tab of the **Table Settings** window.

  You can also define a global transformation to rename the table tablespace on the target. For more information, see Selecting the Transformation Type.

  When this option is not selected, the target tables are created in the default tablespace.

Internal Parameters

Internal parameters are parameters that are not exposed in the UI. You should only use them if instructed by Attunity Support.

To add internal Attunity Replicate parameters:

1. Click the **Internal Parameters** link.

   The **Internal Parameters** dialog box opens.
2. In the edit box, type the name of the parameter you need to add and then click it.
3. The parameter is added to the table below the search box with its default value.
4. Change the default value as required.
5. To reset the parameter value to its default, click the "Restore default value" icon at the end of the row.

Settings Summary

You can view a summary of your settings by clicking the **Setting Summary** link. This is useful if you need to send a summary of your settings to Attunity Support.

Finding the Wallet Entry used for TDE Column Encryption in a Specific Table

This section describes how to find the correct encryption key used for TDE column encryption in a specific table.

To find the Oracle Wallet entry:

1. On the Oracle database, run the following query to return the `object_id` (e.g. the. table ID) according to a given owner and table name:

   ```
   Select object_id from all_objects where owner='<table owner>' and object_name='<table name>' and object_type='TABLE';
   ```
2. Use the retrieved `object_id` in the following query to return the relevant master key:

   ```
   select mkeyid from sys.enc$ where obj#=OBJECT_ID;
   ```
3. Select the key value from the Oracle Wallet as follows:

```
mkstore -wrl <full_wallet_name> -viewEntry <entry_name>
```

> **Note** For more information, see Step 5 in Replicate tablespace names: Select this option to replicate the source tablespace names to an Oracle target. When this option is selected, the Table tablespace name will also be shown in the General tab of the Table Settings window..

4. Copy the master key entry and its value into the **Names** and **Values** fields respectively.

Setting Advanced Connection Properties Using Binary Reader

This section describes which properties are available in the **Advanced** tab when using Binary Reader to access the redo logs. For information on which properties are available in the **Advanced** tab when using LogMiner to access the redo logs, see Setting Advanced Connection Properties Using LogMiner.

> **Note** If your Oracle database version precedes 10.2.0.4 (i.e. version 10.1.x to 10.2.0.3), you must use LogMiner to access the redo logs.

- **Automatically add supplemental logging**: Select this to automatically set up supplemental logging for the Oracle database. This option is also available when LogMiner is selected as the redo logs access method.

  For more information on supplemental logging, see Setting up Supplemental Logging.
- Under the **Access redo logs via** label, choose **Binary Reader**. Replicate will access the redo logs as a binary file.
- **Secret Store encryption entries:** When the source tables are encrypted or contain encrypted columns, you need to specify the Oracle Wallet encryption keys and their values.

  For information on locating the required keys, see Replicate tablespace names: Select this option to replicate the source tablespace names to an Oracle target. When this option is selected, the Table tablespace name will also be shown in the General tab of the Table Settings window..
- **ASM Parameters (if redo logs are stored in ASM)** - If the Oracle redo logs you are using are stored using Automated Storage Management (ASM), enter the required access information in the designated fields.

  > **Note** To access the redo logs in ASM, you also need to grant the additional privileges described in Required ASM Privileges
  >
  > See also: Best Practices when Working with Oracle ASM.

  - **ASM Connection String**: The connection string to the ASM instance if your Oracle database is using ASM.

- **ASM user name**: The user name for the ASM user.
- **ASM password**: The password for the ASM user.
- To access a redo log as a binary file (i.e. *not* using Use LogMiner), select one of the following options:
  - **Use path as it appears in the database:** Select this to access the redo logs using the path as it appears in the database. Continue from Using the Path as it Appears in the Database.

    -OR-
  - **Replace path prefix**: You can determine whether to read the redo logs from a different root location while leaving the relative path unchanged. Continue from Replacing the Path with a Prefix.

Using the Path as it Appears in the Database

- **Attunity Replicate has file-level access to the redo log files**: Select this to access and read the redo logs directly from the file system of the local computer where Attunity Replicate is installed.
- **Copy redo logs to a temporary folder**: Select this to copy the redo logs to a temporary folder and then specify the path of the redo logs on the Oracle machine.

  Note When configuring multiple tasks that use the same temporary folder (configured in the Oracle source endpoint), do not select the **Delete processed archived redo log files** option. This is because Replicate uses the *original* archived log names.

  Note When working in a RAC environment, it is strongly recommended to set up a shared folder that is accessible by all the RAC instances. If this is not possible, you need to define a temporary folder with the same name on each of the RAC instances. In addition, you need to define separate Oracle and ASM connection strings for each RAC instance.

  For more information on defining RAC connection strings, see Setting General Connection Properties.

  - **Replicate has file-level access to temporary folder:** Select this to access the archived redo logs directly from the file system of the local computer where Attunity Replicate is installed.
    - **Access archived redo logs in folder:** To enable Attunity Replicate to access the temporary folder (when it has file level access), you need to specify the path to the shared temporary folder on the Oracle machine, e.g. \\my.oracle.box\tempshare.

Note When a stopped task is resumed, Replicate will try to re-copy the currently processed Redo logs. If there are no Redo logs in the specified directory, the task will wait for them to be copied there.

- **Look for missing archived redo logs in folder:** Type the full path to a location from where you want Attunity Replicate to read the archived redo logs if they are not found in the default location. The folder can be located anywhere in the network where Attunity Replicate is located, but be sure that the location is accessible to the Attunity Replicate user.
  - **Replicate has file-level access to the specified folder:** Select this to access and read the archived redo logs directly from the file system of the local computer where Attunity Replicate is installed.
- **Delete processed archived redo log files**: Select this to delete the copied archived redo log files after they have been read. This option requires the following additional permissions for the Replicate user:
  - GRANT SELECT ON DBA_FILE_GROUPS

    Example:

    ```
    GRANT SELECT ON DBA_FILE_GROUPS to nonpriv_user;
    ```
  - GRANT EXECUTE on SYS.DBMS_FILE_GROUP

    Example:

    ```
    GRANT EXECUTE ON SYS.DBMS_FILE_GROUP to nonpriv_user;
    ```
  - EXECUTE DBMS_FILE_GROUP.GRANT_SYSTEM_PRIVILEGE with the system privilege 'MANAGE_FILE_GROUP' for the Replicate user.

    Example:

    ```
    execute DBMS_FILE_GROUP.GRANT_SYSTEM_PRIVILEGE (DBMS_FILE_
    GROUP.MANAGE_FILE_GROUP, 'nonpriv_user', FALSE)
    ```

  Note Verify that another file group is not using the configured temp directory under a different Oracle user.

- **Retry interval**: Use the counter or type the number of seconds that the system waits before resending a query.
- **Archived redo logs destination identifier**: The destination of the archived redo logs. The value should be the same as the `DEST_ID` number in the `V$archived_log` table.

  Note When working with multiple log destinations (`DEST_ID`), you should specify an **Archived redo logs location identifier** that represents archived logs that can be accessed by Replicate. If the **Archived redo logs location identifier** is not specified, Replicate will use the minimal existing `DEST_ID`.

- **Expose NUMBER as**: Select a precision-scale combination, FLOAT or VARCHAR.

Attunity Replicate supports any precision-scale combination supported by Oracle. By default, the NUMBER data type is converted to precision 38, scale 10.

Note If precision is 39 or above, select VARCHAR.

Note The "Expose NUMBER" definition in the Oracle database is used for the NUMBER data type only (without the explicit precision and scale definition).

- **Use archived redo logs only**: When this option is selected, Attunity Replicate will only access the archived redo logs. If the archived redo logs ares stored on ASM only, the Attunity Replicate user needs to be granted the ASM privileges described in Required ASM Privileges.
- **Replicate tablespace names:** Select this option to replicate the source tablespace names to an Oracle target. When this option is selected, the **Table tablespace** name will also be shown in the **General** tab of the **Table Settings** window.

  You can also define a global transformation to rename the table tablespace on the target. For more information, see Selecting the Transformation Type.

  When this option is not selected, the target tables are created in the default tablespace.

Replacing the Path with a Prefix

- **Replace path prefix**: You can determine whether to read the redo logs from a different root location while leaving the relative path unchanged.

  Type the first part of the path to the current location of the redo logs. For example, `C:\OldFolder`.

  You can include one folder or directory level or multiple folders or directories in this field.

  With: Type the name of the folder or prefix to replace the existing prefix that you added in the field above. For example, `C:\NewFolder`.

  Note The examples illustrate how to change the prefix:

  If the redo logs are located in `C:\OldFolder\archive\logs` and you specify `C:\NewFolder` in the **With** field, the redo logs will be read from:

  `C:\NewFolder\archive\logs`

  If the redo logs are located in `C:\replicate\oracle\logs\archive\RedoLogs` and you specify `C:\replicate\oracle\logs` in the **Replace path prefix** field, and `C:\companyName` in the **With** field, then the redo logs will be read from:

  `C:\companyName\archive\RedoLogs`

  In this case, the new folder or directory called `companyName` replaces all of the first three level folders that you included in the **Replace path prefix** field.

- **Apply prefix replacement to online and archived redo logs**: Select this to apply the prefix replacement to the online and archived redo logs.
  - **Replicate has file-level access to the new location**: Select this to access and read the online and archived redo log files directly from the file system of the local computer where Attunity Replicate is installed.
- **Apply prefix replacement to archived redo logs only:** Select this to apply the prefix replacement to the archived redo logs only (and not to the online redo logs).
  - **Replicate has file-level access to the original online location:** Select this to access and read the original online redo log files directly from the file system of the local computer where Attunity Replicate is installed.
  - **Replicate has file-level access to the new archive location**: Select this to access and read the archived redo log files directly from the file system of the local computer where Attunity Replicate is installed.
- **Delete processed archived redo log files**: Select this to delete the copied archived redo log files after they have been read. This option requires the following additional permissions for the Replicate user:
  - GRANT SELECT ON DBA_FILE_GROUPS

    Example:

    ```
    GRANT SELECT ON DBA_FILE_GROUPS to nonpriv_user;
    ```
  - GRANT EXECUTE on SYS.DBMS_FILE_GROUP

    Example:

    ```
    GRANT EXECUTE ON SYS.DBMS_FILE_GROUP to nonpriv_user;
    ```
  - EXECUTE DBMS_FILE_GROUP.GRANT_SYSTEM_PRIVILEGE with the system privilege 'MANAGE_FILE_GROUP' for the Replicate user.

    Example:

    ```
    execute DBMS_FILE_GROUP.GRANT_SYSTEM_PRIVILEGE (DBMS_FILE_
    GROUP.MANAGE_FILE_GROUP, 'nonpriv_user', FALSE)
    ```

    Note Verify that another file group is not using the configured temp directory under a different Oracle user.

- **Retry interval**: Use the counter or type the number of seconds that the system waits before resending a query.
- **Archived redo logs destination identifier**: The destination of the archived redo logs. The value should be the same as the `DEST_ID` number in the `V$archived_log` table.

Note When working with multiple log destinations (`DEST_ID`), you should specify an **Archived redo logs location identifier** that represents archived logs that can be accessed by Replicate. If the **Archived redo logs location identifier** is not specified, Replicate will use the minimal existing `DEST_ID`.

- **Expose NUMBER as**: Select a precision-scale combination, FLOAT or VARCHAR. Attunity Replicate supports any precision-scale combination supported by Oracle. By default, the NUMBER data type is converted to precision 38, scale 10.

  Note If precision is 39 or above, select VARCHAR.

  Note The "Expose NUMBER" definition in the Oracle database is used for the NUMBER data type only (without the explicit precision and scale definition).

- **Use archived redo logs only**: When this option is selected, Attunity Replicate will only access the archived redo logs. If the archived redo logs ares stored on ASM only, the Attunity Replicate user needs to be granted the ASM privileges described in Required ASM Privileges.
- **Replicate tablespace names:** Select this option to replicate the source tablespace names to an Oracle target. When this option is selected, the **Table tablespace** name will also be shown in the **General** tab of the **Table Settings** window.

  You can also define a global transformation to rename the table tablespace on the target. For more information, see Selecting the Transformation Type.

  When this option is not selected, the target tables are created in the default tablespace.

Internal Parameters

Internal parameters are parameters that are not exposed in the UI. You should only use them if instructed by Attunity Support.

To add internal Attunity Replicate parameters:

1. Click the **Internal Parameters** link.

   The **Internal Parameters** dialog box opens.
2. In the edit box, type the name of the parameter you need to add and then click it.
3. The parameter is added to the table below the search box with its default value.
4. Change the default value as required.
5. To reset the parameter value to its default, click the "Restore default value" icon at the end of the row.

Settings Summary

You can view a summary of your settings by clicking the **Setting Summary** link. This is useful if you need to send a summary of your settings to Attunity Support.

Finding the Wallet Entries used for TDE Tablespace Encryption or TDE Column Encryption

In order to specify the correct encryption key(s) used for TDE tablespace encryption, you first need to find the relevant entry (or entries in the case of multiple keys) in the Oracle Wallet containing the encryption key(s). After you find the relevant entry or entries, copy the entry and its value (or entries and values if more than one) into the **Names** and **Values** fields respectively.

Note To enter multiple values, first copy each entry into a text editor such as Notepad making sure to separate the values with a comma. Then, copy the string containing the values and commas from the text editor and paste it into the **Values** field. There is no need to do this for entries. You can paste the entries directly into the **Entries** field, remembering to separate each entry with a comma.

To find the Oracle Wallet entries:

1. If the ENCRYPTION_WALLET_LOCATION parameter is defined in the sqlnet.ora file, use the wallet from the directory defined by this parameter.
2. If the WALLET_LOCATION parameter is defined in the sqlnet.ora file, use the wallet from the directory defined by this parameter.
3. In other cases, use the wallet in the default database location.

   Note The name of the wallet should be **ewallet.p12**

4. Use the "list" option in the Oracle mkstore utility to determine the ORACLE.SECURITY.DB/TS.ENCRYPTION.<SUFFIX> entry name(s), as follows:

   ```
   mkstore –wrl <full wallet name> -list
   ```

5. If you know which entry/entries is/are used to encrypt the Redo logs, select the entry name(s) and use the "viewEntry" option in the Oracle mkstore utility to determine the entry value, as follows:

   ```
   mkstore –wrl <full wallet name> -viewEntry <entry name>
   ```

   Note If you do not know which entry is used to encrypt the Redo logs, you can select multiple DB or TS entries and determine their values as described above (and then copy and paste the entry names and values into the **Names** and **Values** fields as described in the Setting Advanced Connection Properties Using Binary Reader above). If the specified entries are not correct, the task will fail and the error message will contain the correct entry name.

   Note If the DBA changes the entry while the task is running, the task will fail and the error message will contain the new entry name. Add the new entry (name and value) to the already specified entries and then resume the task.

Required ASM Privileges

The following section describes the additional permissions that are required when the redo logs are stored in ASM.

Grant the following read privilege:

```
SELECT ON v_$transportable_platform
```

From Oracle 11g Release 2 (11.2.0.2), Attunity Replicate must be granted the SYSASM privilege in order to access the ASM account. For older supported versions, granting Attunity Replicate the SYSDBA privilege should be sufficient.

Note When connecting to ASM, Attunity Replicate will first try to log in as SYSDBA and, if unsuccessful, will try to log in as SYSASM.

You can validate ASM account access by opening a command prompt and issuing the following statements:

```
sqlplus asmuser/asmpassword@+asmserver as sysdba
```

-OR-

```
sqlplus asmuser/asmpassword@+asmserver as sysasm
```

Using Microsoft SQL Server as a Source

This section describes how to set up and use a Microsoft SQL Server database as the source database in a replication task.

In this section:

Supported Editions

Attunity Replicate supports the following Microsoft SQL Server editions:

- Enterprise Edition
- Standard Edition
- Workgroup Edition
- Developer Edition

Prerequisites

Make sure the following prerequisites have been met:

- Client prerequisites (for source and target endpoints):

  Attunity Replicate for Windows:

  For all versions of Microsoft SQL Server, Microsoft SQL Server Native Client 11.0 must be installed on the Attunity Replicate Server machine.

  Attunity Replicate for Linux:

  First, install Microsoft ODBC Driver 13.1 for Linux on the Attunity Replicate Server machine.

  Then, on the Attunity Replicate Server machine, open a Unix shell and perform the following steps:

1. Change the working directory to:
   `cd <product_dir>/bin`
2. Stop the Replicate service:
   `/arep.ctl stop`
3. Optionally, confirm that the service has stopped:
   `ps -ef | grep repctl`
4. Create the `site_arep_login.sh` file:
   `touch site_arep_login.sh`
5. Copy the driver location to the `site_arep_login.sh` file:
   `echo "export LD_LIBRARY_PATH=$LD_LIBRARY_ PATH:/opt/microsoft/msodbcsql/lib64/" > site_arep_login.sh`
6. Optionally, confirm that the driver location was copied:
   `cat site_arep_login.sh`
7. Start the Replicate service:
   `./arep.ctl start`
8. Optionally confirm that the service has started:
   `ps -ef | grep repctl`

Note Replicate requires the following ODBC library:

`libmsodbcsql-13.1.so.0.0`

If the existing library has a different version number (e.g. `libmsodbcsql-13.1.so.0.2`), you need to create a symbolic link between the existing library and the required library.

To check which library version is currently installed

Issue the following command:

`cd /opt/microsoft/msodbcsql/lib64/`

To create a symbolic link

Issue the following command:

`ln -s <existing_library_name> libmsodbcsql-13.1.so.0.0`

where `<existing_library_name>` is the name of the currently installed library (e.g. `libmsodbcsql-13.1.so.0.2`).

- A Microsoft SQL Server account with the specific access privileges is required. See Source Permissions or Target Permissions for more information.
- Microsoft SQL Server as a source must be configured for a full backup to work with Attunity Replicate. For more information, see Preparing Microsoft SQL Server Backup and Recovery.

Limitations

When using a Microsoft SQL Server source endpoint in a Replicate task, the following imitations apply:

- A Secondary SQL Server database is not supported as a source database.
- If you are using a Microsoft SQL Server source database in a replication task, the Microsoft SQL Server Replication Publisher definitions for the database that was used in the task are not removed when you remove a task. A Microsoft SQL Server system administrator must delete these definitions from Microsoft SQL Server.
- Sparse tables are not supported.
- Renaming tables using `sp_rename` is not supported (e.g. `sp_rename 'Sales.SalesRegion', 'SalesReg;)`
- Renaming columns using `sp_rename` is not supported (e.g. `sp_rename 'Sales.Sales.Region', 'RegID', 'COLUMN';)`
- TRUNCATE events will not be captured.
- Changes to computed fields in a Microsoft SQL Server source will not be replicated.
- Microsoft SQL Server partition switching is not supported.
- When using the WRITETEXT and UPDATETEXT utilities, Attunity Replicate does not capture events applied on the source database.
- The following DML pattern is not supported:

  ```
  select <*> into <new_table> from <existing_table>
  ```
- Column-level encryption is not supported.
- Due to a known issue with Microsoft SQL Server 2008/2008 R2, Attunity Replicate does not support server level audits on Microsoft SQL Server 2008/2008 R2 as a source database.

  For example, running the following command:

  ```
  USE [master]
  GO
  ALTER SERVER AUDIT [my_audit_test-20140710] WITH (STATE=on)
  GO
  ```

  Will cause Attunity Replicate to fail.

  Note This issue was resolved in Microsoft SQL Server 2012.

- The following limitations apply when accessing the backup transaction logs:
  - Encrypted backups are not supported.
  - Backups stored at a URL or on Windows Azure are not supported.
- The following limitations apply when accessing the backup transaction logs *at file level*:
  - The backup transaction logs must reside in a shared folder with the appropriate permissions and access rights.

- Active transaction logs are accessed through the Microsoft SQL Server API (and not at file-level).
- The Attunity Replicate and Microsoft SQL Server machines must reside in the same domain.
- Compressed backup transaction logs are not supported.
- Transparent Data Encryption (TDE) is not supported. Note that when accessing the backup transaction logs using SQL Server's native functionality (i.e. not using file-level access), TDE encryption *is* supported.
- Unix platforms are not supported.
- Reading the backup logs from multiple stripes is not supported.

For more information on configuring Attunity Replicate to access the backup transaction logs at file-level access, see Setting Advanced Connection Properties.

- Microsoft SQL Server backup to multiple disks is not supported.
- When inserting a value into SQL Server spatial data types (GEOGRAPHY and GEOMETRY), one can either ignore the SRID (Spatial Reference System Identifier) property - in which case the default SRID will be used (0 for GEOMETRY and 4326 for GEOGRAPHY) - or specify a different number. When replicating tables with spatial data types, Attunity Replicate replaces the SRID that was inserted by user with the default SRID.
- If your database is not set up for MS-REPLICATION or MS-CDC, you can still capture tables that do not have a Primary Key, but bear in mind that in such a setup only INSERT/DELETE DML events will be captured. UPDATE and TRUNCATE TABLE events will be ignored.
- Columnstore indexes are not supported.
- Memory-optimized tables (using In-Memory OLTP) are not supported.
- When replicating a table with a Primary Key that consists of multiple columns, updating the Primary Key columns during Full Load is not supported.
- Temporal databases are not supported
- Delayed durability is not supported
- Table change tracking is not supported

Non-Supported Microsoft SQL Server Security Features

Tables that use the following Microsoft SQL Server security features are not supported:

- Always Encrypted
- Dynamic Data Masking
- Row-Level Security

Working with Microsoft SQL Server AlwaysOn Availability Groups

The Microsoft SQL Server AlwaysOn Availability Groups feature is a high-availability, disaster-recovery solution that provides an enterprise-level alternative to database mirroring.

Note Support for AlwaysOn Availability Groups is not available when Replicate is installed on Linux.

Accessing Backup Logs in AlwaysOn Availability Groups

As opposed to active transaction logs which are synchronized across the AlwaysOn Availability Group, backup transaction logs are local to each individual replica.

Each of the replicas (primary or secondary) in an AlwaysOn Availability Group can create local backup logs. However, since Replicate should be configured to connect to the Availability Group Listener (see below) which routes the connection to a *primary* replica, it has no way of determining whether the secondary replica's backup logs are present and/or accessible. This can result in Replicate missing events that are only present in the backup logs on the secondary replica.

To prevent this from happening, you need define the backup maintenance plan for the AlwaysOn configuration so that only *one of the replicas maintains the backup logs*. This is the replica that should be specified in the **AlwaysOn backup replica** field described below.

For a detailed explanation of how to set up a maintenance plan, see https://msdn.microsoft.com/en-us/library/ms191002.aspx

Important: The maintenance plan must be performed *directly* on the designated backup replica.

In the event that Replicate is connected to a primary replica that is not the AlwaysOn backup replica and needs to access the backup transaction logs, it will be unable to do so. In such a situation, the task will stop with a fatal error.

The scenarios where Replicate needs to access the backup transaction logs are as follows:

- Working in backup only mode.

  For more information on this mode, see Read changes from backup only.
- Starting a task from a specific timestamp.

  For more information on this option, see the Tables are already loaded in Using Advanced Run Options.
- Due to latency i.e. if there is a high rate of events (changes) that Replicate is unable to process using the active log only.

Note Replicate relies on backup maintenance plan being implemented as described above. Deviating from this plan may result in lost events if Replicate needs to access the backup logs.

To use AlwaysOn Availability Groups in Attunity Replicate:

1. Enable Distribution on all Microsoft SQL Server instances in your Availability Replicas.

   Note This is required regardless of whether you are working in MS-REPLICATION or MS-CDC mode. For more information on working in MS-CDC mode, see Replicating Tables that do not have a Primary Key.

2. In Replicate:
   a. Open the Microsoft SQL Server endpoint (source or target) settings.
   b. In the **Server Name** field, specify the DSN name or IP address that was configured for the Availability Group Listener.
   c. Specify the name of the replica server where Replicate expects to find the backup logs in the **AlwaysOn backup replica** field in the **Advanced** tab of the Microsoft SQL Server source endpoint. The name must be specified without the domain suffix, even if it is the correct domain. For example, instead of `myreplica.qa.int`, specify `myreplica`.

Note

- When you start a Replicate task for the first time, it may take longer than usual to start as the creation of the table articles is being duplicated by the Availability Groups Server.
- In the event of failover, you will see recoverable errors in the log and the task will restart. This is normal as the connection is being forcibly closed by the failover. As soon as the new primary server is available, processing will resume.

Using a Microsoft SQL Server Endpoint as a Source

The following topics describe what you need to use a Microsoft SQL Server source in an Attunity Replicate task:

- Supported Compression Methods
- Required Permissions
- Microsoft SQL Server Source Data Types
- Non-Supported Data Types
- Preparing the Microsoft SQL Server Database for Replication
- Setting General Connection Properties

Required Permissions

To use a Microsoft SQL Server source in an Attunity Replicate task, the user specified in the Microsoft SQL Server endpoint connection settings must have the **db_owner** database role membership and be granted the following database privileges:

Note Before granting the permissions, make sure that the Replicate user is also mapped to both the MSDB and MASTER databases.

- SELECT ON fn_dblog

  Allows the user to access the active TLOG with API.
- VIEW SERVER STATE

  Allows the user to view the AlwaysOn configuration details.
- EXECUTE on msdb.dbo.sp_stop_job

  Allows the task/user to control the truncation safeguard mechanism.
- EXECUTE on msdb.dbo.sp_start_job

  Allows the task/user to control the truncation safeguard mechanism.
- SELECT

  Allows the task to view the backup history.

Note If you do not need to grant *minimum* permissions, you can grant the user the `sysAdmin` fixed server role on the Microsoft SQL Server database instead.

Supported Compression Methods

The table below lists which compression methods Attunity Replicate supports for each Microsoft SQL Server version.

Table 8.4 | Supported Microsoft SQL Server Compression Methods

Microsoft SQL Server Version	Row/Page Compression (at Partition Level)	Vardecimal Storage Format	Vardecimal Storage Format Sparse Columns	Vardecimal Storage Format Sparse Columns Columnar Structure Compression
2005	No	No	No	No
2008	Yes	No	No	No
2012	Yes	No	No	No

Microsoft SQL Server Source Data Types

The Microsoft SQL Server source for Attunity Replicate supports most Microsoft SQL Server data types. The following table shows the Microsoft SQL Server source data types that are supported when using Attunity Replicate and the default mapping to Attunity Replicate data

types. Note that Microsoft SQL Server data types are only mapped to Attunity Replicate data types when the target endpoint is *not* Microsoft SQL Server. For information on data type mapping and collation support when the target endpoint *is* Microsoft SQL Server, see Homogeneous Replication below.

For information on how to view the data type that is mapped in the target, see the section for the target endpoint you are using.

For additional information about Attunity Replicate data types, see Replicate Data Types.

Note Collatable data types are indicated by an asterisk (*).

Table 8.5 | Microsoft SQL Server Source Data Types with Mapping to Attunity Replicate Data Types when the Target is not Microsoft SQL Server

Microsoft SQL Server Data Types	Attunity Replicate Data Types
BIGINT	**INT8**
BIT	**BOOLEAN**
DECIMAL	**NUMERIC**
INT	**INT4**
MONEY	**NUMERIC (19,4)**
NUMERIC (p,s)	**NUMERIC**
SMALLINT	**INT2**
SMALLMONEY	**NUMERIC (10,4)**
TINYINT	**UINT1**
REAL	**REAL4**
FLOAT	**REAL8**
DOUBLE	**REAL8**
DATETIME	**DATETIME**
DATETIME2 (Microsoft SQL Server 2008 and later)	**DATETIME**
SMALLDATETIME	**DATETIME**
DATE	**DATE**
TIME	**STRING (16)**
DATETIMEOFFSET	**STRING**
*CHAR	**STRING**
*VARCHAR	**STRING**

Table 8.5 | Microsoft SQL Server Source Data Types with Mapping to Attunity Replicate Data Types when the Target is not Microsoft SQL Server (Cont.)

Microsoft SQL Server Data Types	Attunity Replicate Data Types
*VARCHAR (max) *TEXT To use this data type with Attunity Replicate, you must enable the use of CLOBs for a specific task. LOB columns for Microsoft SQL Server tables are updated in the target even for UPDATE statements that did not change the value of the LOB column in Microsoft SQL Server. During CDC, CLOB data types are supported only in tables that include a primary key. For more information, see **LOB support** in **Task Settings**/Metadata.	**CLOB**
*NCHAR	**WSTRING**
*NVARCHAR (length)	**WSTRING**
*NVARCHAR (max) *NTEXT To use this data type with Attunity Replicate, you must enable the use of NCLOBs for a specific task. LOB columns for Microsoft SQL Server tables are updated in the target even for UPDATE statements that did not change the value of the LOB column in Microsoft SQL Server. During CDC, NCLOB data types are supported only in tables that include a primary key. For more information, see **LOB support** in **Task Settings**/Metadata.	**NCLOB**
BINARY	**BYTES**
VARBINARY	**BYTES**
VARBINARY (max) IMAGE LOB columns for Microsoft SQL Server tables are updated in the target even for UPDATE statements that did not change the value of the LOB column in Microsoft SQL Server. To use this data type with Attunity Replicate, you must enable the use of BLOBs for a specific task. BLOB data types are supported only in tables that include a	**BLOB**

Table 8.5 | Microsoft SQL Server Source Data Types with Mapping to Attunity Replicate Data Types when the Target is not Microsoft SQL Server (Cont.)

Microsoft SQL Server Data Types	Attunity Replicate Data Types
primary key. For more information, see **LOB support** in **Task Settings**/Metadata.	
TIMESTAMP	**BYTES**
UNIQUEIDENTIFIER	**STRING**
HIERARCHYID	**HIERARCHYID** - When replicating to Microsoft SQL Server. **STRING (250)** - When replicating to all other endpoints.
XML LOB columns for Microsoft SQL Server tables are updated in the target even for UPDATE statements that did not change the value of the LOB column in Microsoft SQL Server. To use this data type with Attunity Replicate, you must enable the use of NCLOBs for a specific task. During CDC, NCLOB data types are supported only in tables that include a primary key. For more information, see **LOB support** in **Task Settings**/Metadata.	**CLOB**
GEOMETRY	**CLOB**
GEOGRAPHY	**CLOB**

Non-Supported Data Types

Tables that include fields with the following data types are not supported by Attunity Replicate.

- CURSOR
- SQL_VARIANT
- TABLE

Note User-defined data types are supported according to their base-type. For example a user-defined data type based on DATETIME is handled as a DATETIME data type.

Homogeneous Replication

When replicating from a Microsoft SQL Server source to a Microsoft SQL Server target, most of the source and target data types will be identical. The exceptions are listed in the table below.

Note In homogeneous replication, the source data first passes through the Attunity Replicate data type and is therefore subject to any limitations of that type.

For information on Replicate data types and their limitations (where relevant), see Replicate Data Types.

For information on which Replicate data types the source data passes through when replicating from Microsoft SQL Server, see the Microsoft SQL Server to Attunity Replicate data types mapping table described earlier.

Note To prevent data truncation when replicating **XML**, **Geometry** and **Geography** data types, it is strongly recommended to enable the Allow unlimited LOB size option in the task settings.

Additionally, in homogeneous replication, source column and table collations will be replicated to the target as described in Column and Table Collation.

Data Type Exceptions

When replicating from one Microsoft SQL Server database to another, source and target data types are identical for all supported Microsoft SQL Server versions, with the following exceptions:

Table 8.6 | Data Type Exceptions in Homogeneous Replication

Microsoft SQL Server Source	Microsoft SQL Server 2005 Target	Microsoft SQL Server 2005-2016 Target
DATETIME2 (Microsoft SQL Server 2008 and later)	DATETIME (when prec<=3) else VARCHAR (37)	
DATE	VARCHAR (11)	
TIME	VARCHAR (27)	
DATETIMEOFFSET	VARCHAR (34)	
VARCHAR		VARCHAR (x) (when x=0 or x>8000) else VARCHAR (max)
NVARCHAR (length)		NVARCHAR (x)

Table 8.6 | Data Type Exceptions in Homogeneous Replication (Cont.)

Microsoft SQL Server Source	Microsoft SQL Server 2005 Target	Microsoft SQL Server 2005-2016 Target
		(when x=0 or x>8000) else NVARCHAR (max)
VARBINARY		VARBINARY (x) (when x=0 or x>8000) else VARBINARY (max)
HIERARCHYID		VARCHAR (x)
GEOMETRY	VARCHAR (MAX)	
GEOGRAPHY	VARCHAR (MAX)	
TIMESTAMP		VARBINARY

Column and Table Collation

When replicating from one Microsoft SQL Server database to another, column and table collations will be replicated to the target.

Note To support collation replication, the DBA must ensure that the collations defined for the source Microsoft SQL Server database are the same as those defined for the target Microsoft SQL Server database.

Non-Nullable Columns and Primary/Unique Index Names

Primary/Unique Index names are preserved during homogeneous replication. Non-nullable columns are also preserved during homogeneous replication, with the exception of the following data types:

- text
- ntext1
- varchar(max)
- nvarchar(max)
- varbinary(max)
- image
- xml

Preparing the Microsoft SQL Server Database for Replication

This topics describes the configuration requirements for using a Microsoft SQL Server database. A Microsoft SQL Server system administrator should carry out these tasks.

Preparing Microsoft SQL Server Backup and Recovery

Attunity Replicate consumes changes captured from the database transaction log (TLOG). The TLOG is maintained by Microsoft SQL Server for recovery purposes. All changes made to a database are written to the TLOG. The following happens when recovery is required:

- A backup copy of the database is made.
- Logged events are taken and used in a rolling-forward process where the recorded changes are replayed against that copy.

To prepare for backup and recovery you must make sure that the Microsoft SQL Server Recovery Model is set up. You select the Recovery Model in the Microsoft SQL Server Management Studio. This should be carried out by a Microsoft SQL Server system administrator.

The TLOG data is truncated as soon as it is no longer needed therefore the TLOG is not persistent. However, Attunity Replicate guaranteed delivery requires persistency in the changed data. To ensure persistency:

- A full database backup must be carried out before beginning to replicate data.
- The Recovery Model must be set to **Bulk logged** or **Full**.

To set the recovery model:

In the database properties **Options** tab, set the Recovery Model to **Bulk logged** or **Full**. In these modes, the transaction Log is more durable.

Setting up Microsoft SQL Server for Replication

If you are using Microsoft SQL Server as the source in an Attunity Replicate task, you need to enable your Microsoft SQL Server database for MS-REPLICATION.

In the Microsoft SQL Server's Management Studio, follow the instructions provided by the Configure Distribution wizard to set up replication or see the Microsoft SQL Server documentation.

To open the wizard from Microsoft SQL Server:

1. In the Microsoft SQL Server Management Studio, right-click the Replication folder and select **Configure Distribution**.

   The Configure Distribution wizard opens.
2. Make the following selections:

   In the Distributor step, select **<Microsoft SQL Server Name> will act as its own distributor; Microsoft SQL Server will create a distribution database and log**.

Replicating Tables that do not have a Primary Key

Note This functionality is supported only for Microsoft SQL Server Enterprise edition and *not* on Microsoft SQL Server 2005.

By default, Attunity Replicate automatically sets up MS-REPLICATION for each of the source tables in a replication task. However, MS-REPLICATION requires each of the source tables to have a primary key, which may not always be the case. Therefore, if you need to replicate tables that do not have a primary key, the following options are available:

- Use MS-CDC
- Do not Use MS-Replication or MS-CDC

Use MS-CDC

To set up MS-CDC, you first need to enable MS-CDC for the database by running the following command:

```
use [DBname]
EXEC sys.sp_cdc_enable_db
```

Then you need to enable MS-CDC for each of the source tables by running the following command:

```
EXECUTE sys.sp_cdc_enable_table @source_schema = N'MySchema', @source_name
= N'MyTable', @role_name = NULL;
```

Note Replicating tables that do not have a Primary Key or a Unique Index may adversely affect performance (since additional database resources are required to capture the changes). However, you can prevent performance issues related to the absence of Primary Keys or a Unique Index by manually adding indexes to the target tables.

For more information on setting up MS-CDC for specific tables, please refer to the Microsoft website.

Do not Use MS-Replication or MS-CDC

If your database is not set up for MS-REPLICATION or MS-CDC, you can still capture tables that do not have a Primary Key, but bear in mind that in such a setup only INSERT/DELETE DML events will be captured. UPDATE and TRUNCATE TABLE events will be ignored.

It is also important to note that a DELETE statement executed on an UPDATED source record, will *not* be applied on the target.

Defining Microsoft SQL Server Database Settings

Set the following for the Microsoft SQL Server database(s) that you are using as a source:

- From the Object Explorer in the Microsoft SQL Server Management Studio, right click the database and select **Properties**. In the **Options** tab, set the **Recovery model** to **Bulk logged** or **Full**. In this mode, the transaction Log is more durable and truncation occurs less frequently.
- Ensure that there is a full database backup for each Microsoft SQL Server database that you are using as a source.
- When creating a connection string, it is possible to use any parameter supported by Microsoft SQL Server. The Microsoft SQL Server system administrator must ensure that the Microsoft SQL Server instance is configured correctly so that the proper authentication credentials are accepted.
- To be able to work with MS-REPLICATION, each of the source tables must have a primary key.

Working with Windows Authentication

You can configure the Attunity Replicate Microsoft SQL Server endpoint to log in to Microsoft SQL Server (on Windows) using Windows authentication.

If you choose this option, you also need to make sure that:

- The Microsoft SQL Server instance is set up to allow Windows log on.
- The Attunity Replicate user is specified as the "Log on as" user for the "Attunity Replicate Server" service account.

  -OR-

  Microsoft SQL Server is configured to allow login for the Attunity Replicate Server service account.

Setting General Connection Properties

This section describes how to configure general connection properties. For an explanation of how to configure advanced connection properties, see **Setting Advanced Connection Properties** below.

To add a Microsoft SQL Server source endpoint to Attunity Replicate:

1. In the Attunity Replicate Console, click Manage Endpoint Connections to open the **Manage Endpoints Connections** dialog box. Then click the **New Endpoint Connection** button. For more information on adding an endpoint to Attunity Replicate, see Working with Endpoints.
2. In the **Name** field, type a name for your database. This can be any name that will help to identify the database being used.
3. In the **Description** field, type a description that helps to identify the Microsoft SQL Server database. This is optional.
4. Select **SOURCE** as the database **role**.

5. Select **Microsoft SQL Server** as the database **Type**.
6. Specify the **Server name**. This is the host name or IP address of the computer with the Microsoft SQL Server instance containing the source database.

   Note: To override the default port, add the port to the server name, separated by a comma. For example, if the server name is **`myserver.company.local`** and the port is **`3333`**, then the server name should be entered like this:

   `myserver.company.local,3333`
7. Select **Windows authentication** (only relevant when Replicate is installed on Windows) or **SQL Server authentication**.

   If you select **Windows authentication**, the user credentials for the Windows domain will be used. This privilege must be configured in the Microsoft SQL Server database by the system administrator. Note that this option is not relevant when Microsoft SQL Server is running on Linux.

   Note When using **Windows authentication**, make sure that the user account that is associated with the Attunity Replicate **Server** service has Network read and write permissions. This must be configured by a Windows system administrator.

   See also Working with Windows Authentication.

   If you select **SQL Server authentication**, type the Microsoft SQL Server authentication information (User name, Password) for the authorized user for this Microsoft SQL Server database. If you do not know this information, see the Microsoft SQL Server System Administrator.

Configuring Replicate to Automatically Replace the User-Entered Password

To prevent illicit database activity by unauthorized third-parties, Replicate can be configured to automatically replace the user-entered password with a strong random password.

Note This feature cannot be used when the user name is "sa".

Note Clicking the "Test Connection" button will verify the connection using the original password. The password will be automatically changed the first time the task runs.

To utilize this feature, the password must be defined both in the Replicate endpoint settings *and* on the actual database, in the following format:

`replace:`*`your_password`*

Example:

`replace:k$RJdg7!S&ib`

Defining Multiple Endpoints to use the same Automatically Changed Password

In Attunity Replicate, more than one endpoint may be configured to connect to the same database server.

To allow multiple endpoints to use the same (automatically changed) credentials, the password in one of the endpoints needs to defined. Then, each of the other endpoint connections needs to be configured to reference that endpoint.

Note A source endpoint cannot reference a target endpoint, only another source endpoint. Likewise, a target endpoint cannot reference a source endpoint, only another target endpoint.

Additionally, an endpoint cannot reference another endpoint that uses a different database server.

To configure an endpoint to use the automatically changed credentials of another endpoint:

a. In the **User name** field, enter the user name in the following format:

   `ref:endpoint_name`

   Where `endpoint_name` is the name of the endpoint connection whose password was automatically changed.

b. In the **Password** field, specify the password *before* it was automatically changed and without the "`replace`" prefix.

   Example:

   If the original password is:

   `replace:54lakrfgnier3!`

   Specify:

   `54lakrfgnier3!`

Note

- This information is case sensitive.
- To determine if you are connected to the database you want to use or if the connection information you entered is correct, click **Test Connection**.

  If the connection is successful a message in green is displayed. If the connection fails, an error message is displayed at the bottom of the dialog box.

  To view the log entry if the connection fails, click **View Log**. The server log is displayed with the information for the connection failure. Note that this button is not available unless the test connection fails.

Important: Make sure that the Microsoft SQL Server user has the correct access privileges. For information on how to provide the required privileges, see Required Permissions.

8. Type the **Database name** or click **Browse** and select one from the list of available databases. This is the name of the database from where you are replicating the data.

Setting Advanced Connection Properties

In the **Advanced** tab, you can set the following properties:

- **Prevent truncation of unread changes from TLOG:** For optimal performance, Attunity Replicate will try to capture all unread changes from the active transaction log (TLOG). However, sometimes due to truncation, the active TLOG may not contain all of the unread changes. When this occurs, Attunity Replicate accesses the backup log to capture the missing changes. To minimize the need to access the backup log, Attunity Replicate prevents truncation using one of the following methods:
  - **Start transactions in the database:** This is the default method. When this method is used, Attunity Replicate prevents TLOG truncation by mimicking a transaction in the database. As long as such a transaction is open, changes that appear after the transaction started will not be truncated. If you need Microsoft Replication to be enabled in your database, then you must choose this method.

    Note When this option is selected, Replicate creates a table named `attrep_truncation_safeguard` in the source database. This is a very small but important table whose purpose is to prevent truncation of the transaction log by mimicking a transaction in the database. The table can be safely deleted if there are no tasks configured with the **Start transactions in the database** option.

  - **Exclusively use sp_repldone within a single task:** When this method is used, Attunity Replicate reads the changes and then uses `sp_repldone` to mark the TLOG transactions as ready for truncation. Although this method does not involve any transactional activities, it can only be used when Microsoft Replication is not running. Also, using this method, only one Attunity Replicate task can access the database at any given time. Therefore, if you need to run parallel Attunity Replicate tasks against the same database, use the default method.

    Note This method requires the Log Reader Agent to be stopped in the database. If the Log Reader Agent is running when the task starts, Attunity Replicate will forcibly stop it. Alternatively, you can stop the Log Reader Agent manually, before starting the Attunity Replicate task. For instructions on how to do this, refer to the Microsoft SQL Server Management Studio help.

    Note This option is not available when the Microsoft SQL Server Replication job resides on a remote Distributor machine as Replicate does not have access to the remote machine.

  - **Apply TLOG truncation prevention policy every (seconds):** Specify how often to prevent TLOG truncation using one of the methods describes above. Factors that

you should consider when determining the policy frequency include storage availability, backup and log routines, and the rate at which Attunity Replicate processes events.

- **Alternate backup folder:** The location of the backup logs when using a third-party utility to back up the transaction logs (i.e. instead of Microsoft SQL Server's own backup mechanism). You can run the backup utility yourself or you can configure Attunity Replicate to run it as described in Backup file preprocessing command below.

  Note that the backup files must be exported to the specified location in standard Microsoft SQL Server format.
- **Change processing mode:** Choose one of the following change processing modes:
  - **Read changes from online log** - This is the default. Attunity Replicate first reads the online logs for changes and then reads the backup logs.
  - **Read changes from backup if the same record exists in online and backup logs** - When this option is enabled and the same record appears in both the active log and the backup logs, Attunity Replicate will read the changes from the backup transaction logs instead of the online transaction log. This can improve performance when reading from the online transaction log is slow (e.g due to lock contention) or when using file-level access to access the backup transaction logs.
  - **Read changes from backup only** - When this option is selected, Attunity Replicate will read the changes from the backup transaction logs only. Selecting this method results in increased latency due to the interval between backups. The actual latency time will remain constant, but will vary according to the backup schedule.
- Attunity Replicate **has file-level access to the backup log files:** Select this option if Attunity Replicate has been granted file-level access to the backup log files in the **Alternate backup folder**.

Note When Attunity Replicate has file-level access to the backup transaction logs, the following rules apply:

- The **Alternate backup folder** must be a common shared network folder, for example: \\temp\backup.
- The Attunity Replicate **Server** service must be configured to log on using the user name and password specified in the **Backup folder user name** and **Backup folder password** fields.

  To do this:

  In the Windows **Services** console, double-click the Attunity Replicate **Server** service.

  In the **Log On** tab, select **This account** and then enter the user name and password.
- The specified user must be granted Read permission to the alternate backup folder (i.e. the shared network folder).

For a complete list of the limitations affecting file-level access, see Limitations.

- **Backup folder user name:** The user name required to access the backup folder when Attunity Replicate has file-level access.
- **Backup folder password:** The password required to access the backup folder when Attunity Replicate has file-level access.

- **Backup file preprocessing command:** You can use a third-party utility to convert the transaction logs to standard Microsoft SQL Server format (if they are in a different format) and back them up to an alternate backup folder. This option should be used in conjunction with the Alternate backup folder option described above.

  Prerequisites and Notes:

  The command is invoked via the XP_CMDSHELL extended procedure.

  - The backup utility is responsible for setting the system return code (0 for success, 1 for failure), assuming that this code is delegated as the XP_CMDSHELL return value.
  - The backup utility invoked by XP_CMDSHELL must have the same security rights as the Microsoft SQL Server service account.
  - XP_CMDSHELL is normally disabled. It can be enabled and disabled by using the Policy-Based Management or by executing SP_CONFIGURE.
  - Using this extended procedure requires CONTROL SERVER permission (at least).

  Command Usage:

  The backup utility should provide Attunity Replicate with the following parameters:

  - `{BACKUP_INFILE}` - The full path to the original backed up transaction log.
  - `{ALTDIR_OUTFILE}` - The specifications of the target file to transfer to the alternate backup folder.
  - `{BACKUP_SET}` - The backup set to be processed within the backup log.

  Example command:

  ```
  C:\Temp\YourBackupUtility.exe -B{BACKUP_INFILE} -A{ALTDIR_OUTFILE}"
  ```

  Important: Directory names in the command path or file names in the actual command that contain spaces must be enclosed in double-quotes:

  Example:

  C:\temp\test\"my program"\"new version"\converter.exe -A{"input file"} -B{outfile}

- **Delete processed backup logs**: Select this option to delete the backup logs after they have been read.
- **Select virtual backup device types:** When this option is selected, Attunity Replicate will read changes from the specified virtual device(s). Usually, this option only needs to be enabled when using a third-party backup utility (which will be recorded as a virtual device).
- **AlwaysOn backup replica**: See Working with Microsoft SQL Server AlwaysOn Availability Groups.

Internal Parameters

Internal parameters are parameters that are not exposed in the UI. You should only use them if instructed by Attunity Support.

To add internal Attunity Replicate parameters:

1. Click the **Internal Parameters** link.

   The **Internal Parameters** dialog box opens.
2. In the edit box, type the name of the parameter you need to add and then click it.
3. The parameter is added to the table below the search box with its default value.
4. Change the default value as required.
5. To reset the parameter value to its default, click the "Restore default value" icon at the end of the row.

Settings Summary

You can view a summary of your settings by clicking the **Setting Summary** link. This is useful if you need to send a summary of your settings to Attunity Support.

Using SAP Sybase ASE as a Source

This section describes how to set up and use a SAP Sybase ASE database as the source endpoint in a replication task.

In this section:

Prerequisites

Make sure the following prerequisites have been met:

- SAP Sybase ASE ODBC 64-bit client installed on the computer where Attunity Replicate is located.
- SAP Sybase ASE replication enabled for tables using the `sp_setreptable` command or privileges to enable it automatically.
- `RepAgent` must be disabled on the SAP Sybase ASE database.
- When replicating to SAP Sybase ASE 15.7 installed on a Windows machine configured with a non-Latin language (e.g. Chinese), Attunity Replicate requires Sybase 15.7 SP121 to be installed on the SAP Sybase ASE machine.

Limitations

The following limitations apply:

- Only one Attunity Replicate task can be run per SAP Sybase ASE database.
- Attunity Replicate tasks cannot run concurrently with SAP Sybase ASE Replication Server against the same SAP Sybase ASE database.
- Rename table is not supported (e.g. `sp_rename 'Sales.SalesRegion', 'SalesReg;)`
- Rename column is not supported (e.g. `sp_rename 'Sales.Sales.Region', 'RegID', 'COLUMN';)`
- Zero values located at the end of binary data type strings are truncated when replicated to the target database. For example, `0x00000000000000000000000001000001000000000` in the source table will become `0x00000000000000000000000001000001` in the target table.

- The `reorg rebuild` index command is not supported.
- Clusters are not supported.

Required Permissions

To use SAP Sybase ASE database as a source in a Replicate task, the following permissions are required:

- sa_role
- replication_role
- sybase_ts_role

If the **Automatically enable Sybase replication** option is enabled (in the **Advanced** tab), Replicate also needs permission to run the stored procedure `sp_setreptable`.

For information on the **Automatically enable SAP Sybase ASE replication** option, see Setting Advanced Connection Properties.

SAP Sybase ASE database Source Data Types

The following table shows the SAP Sybase ASE database source data types that are supported when using Attunity Replicate and the default mapping from Attunity Replicate data types.

For information on how to view the data type that is mapped in the target, see the section for the target database you are using.

For additional information about Attunity Replicate data types, see Replicate Data Types.

Table 8.7 | SAP Sybase ASE database Source Data Types with Mapping to Attunity Replicate Data Types

SAP Sybase ASE Source Data Types	Attunity Replicate Data Types
BIGINT	INT8
UNSIGNED BIGINT	UINT8
INT	INT4
UNSIGNED INT	UINT4
SMALLINT	INT2
UNSIGNED SMALLINT	UINT2
TINYINT	UINT1
DECIMAL	NUMERIC
NUMERIC	NUMERIC
FLOAT	REAL8

Table 8.7 | SAP Sybase ASE database Source Data Types with Mapping to Attunity Replicate Data Types (Cont.)

SAP Sybase ASE Source Data Types	Attunity Replicate Data Types
DOUBLE	REAL8
REAL	REAL4
MONEY	NUMERIC
SMALLMONEY	NUMERIC
DATETIME	DATETIME
BIGDATETIME	DATETIME (6)
SMALLDATETIME	DATETIME
DATE	DATE
TIME	TIME
BIGTIME	TIME
CHAR	STRING
UNICHAR	WSTRING
NCHAR	WSTRING
VARCHAR	STRING
UNIVARCHAR	WSTRING
NVARCHAR	WSTRING
BINARY	BYTES
VARBINARY	BYTES
BIT	BOOLEAN
TEXT	CLOB
UNITEXT	NCLOB
IMAGE	BLOB

Non-Supported Data Types

Source SAP Sybase ASE tables with columns of the following SAP Sybase ASE data types cannot be replicated. Replicated columns with these data types will show as null.

- UDT

Setting General Connection Properties

This section describes how to configure general connection properties. For an explanation of how to configure advanced connection properties, see **Setting Advanced Connection**

Properties below.

Note You can also use SAP Sybase ASE files as a source. For more information, see Using the Attunity Replicate File Channel.

To add a SAP Sybase ASE source endpoint to Attunity Replicate:

1. In the Attunity Replicate Console, click **Manage Endpoint Connections** to open the **Manage Endpoints Connections** dialog box. Then click the **New Endpoint Connection** button. For more information on adding an endpoint to Attunity Replicate, see Working with Endpoints.
2. In the **Name** field, type a name for your database. This can be any name that will help to identify the database being used.
3. In the **Description** field, type a description that helps to identify the SAP Sybase ASE database. This is optional.
4. Select **SOURCE** as the endpoint **role**.
5. Select **SAP Sybase ASE** as the database **Type**.
6. In the **Server Name** field, enter the host name or IP address of the computer on which the SAP Sybase ASE database is installed.

   Note Consider the following:

   - This information is case sensitive.
   - You can use the **Advanced** tab to add specific properties and create a custom connect string. In this case, you do not need to enter information in this tab. For more information on using the **Advanced** tab, see Setting Advanced Connection Properties.
   - To determine if you are connected to the database you want to use or if the connection information you entered is correct, click **Test Connection**

     If the connection is successful a message in green is displayed. If the connection fails, an error message is displayed at the bottom of the dialog box.

     To view the log entry if the connection fails, click **View Log**. The server log is displayed with the information for the connection failure. Note that this button is not available unless the test connection fails.

7. Optionally, change the default port (5000).
8. Type the SAP Sybase ASE authentication information (**User Name**, **Password**) for the authorized user for this SAP Sybase ASE database. If you do not know this information, see your SAP Sybase ASE database Administrator (DBA).

Note Consider the following:

- This information is case sensitive.
- This information is required. If you are using the **Advanced** tab to create a custom string, make sure to include the **User Name** and **Password** properties. See Setting Advanced Connection Properties for more information.
- If you want to set custom properties for this database, see Setting Advanced Connection Properties.

Important: Make sure that the SAP Sybase ASE user entered in the SAP Sybase ASE Authentication section has the correct access privileges. For information on how to provide the required privileges, see Required Permissions.

9. In the **Database name** field, enter the SAP Sybase ASE database name.

Setting Advanced Connection Properties

In the **Advanced** tab, you can set the following parameters:

- **Automatically enable SAP Sybase ASE replication**: Select this to automatically enable SAP Sybase ASE replication. This is only required if SAP Sybase ASE replication has not been enabled already. For more information, see Prerequisites.
- **Additional ODBC connection properties**: Specify any additional ODBC connection parameters that you want to use.

Note If the user name or password specified in the **General** tab contains non-Latin characters (e.g. Chinese), the following property is required:

```
charset=gb18030
```

Internal Parameters

Internal parameters are parameters that are not exposed in the UI. You should only use them if instructed by Attunity Support.

To add internal Attunity Replicate parameters:

1. Click the **Internal Parameters** link.

   The **Internal Parameters** dialog box opens.
2. In the edit box, type the name of the parameter you need to add and then click it.
3. The parameter is added to the table below the search box with its default value.
4. Change the default value as required.
5. To reset the parameter value to its default, click the "Restore default value" icon at the end of the row.

Settings Summary

You can view a summary of your settings by clicking the **Setting Summary** link. This is useful if you need to send a summary of your settings to Attunity Support.

Removing the Truncation Point

When a task starts, Replicate establishes a `$replication_truncation_point` entry in the `syslogshold` system view, indicating that a replication process is in progress. While Attunity Replicate is working, it advances the replication truncation point at regular intervals, according to the amount of data that has already been copied to the target.

Once the `$replication_truncation_point` entry has been established, the Replicate task must be kept running at all times to prevent the database log from becoming excessively large. If you want to stop the Replicate task permanently, the replication truncation point must be removed by issuing the following command:

```
dbcc settrunc('ltm','ignore')
```

After the truncation point has been removed, the Replicate task cannot be resumed. The log will continue to be truncated automatically at the checkpoints (if automatic truncation is set).

Using MySQL as a Source

This section describes how to set up and use a MySQL database as the source in a replication task.

Note Attunity Replicate also supports Percona as a source endpoint and MariaDB as both a source and a target endpoint. The procedures for configuring connectivity to these endpoints are identical to those described for MySQL. However, when using Percona as a source, there is no need to perform the procedures described in Cluster Prerequisites.

In this section:

Prerequisites

Make sure that the prerequisites described in this section have been met.

In this section:

General Prerequisites

The following is required:

- Attunity Replicate installed on Windows or Linux in your network.
- A MySQL account with the required Security Requirements.
- A MySQL database with the tables that you want to replicate should be accessible in your network.

  The following MySQL editions are supported:

  - MySQL Community Edition
  - MySQL Standard Edition
  - MySQL Enterprise Edition
  - MySQL Cluster Carrier Grade Edition

Attunity Replicate Server for Windows

To be able to work with Attunity Replicate for Windows and MySQL as a source or target endpoint in a Attunity Replicate task, you need to:

- Install a MySQL ODBC 64-bit client version 5.2.6 or later on the same computer as Attunity Replicate.

Attunity Replicate Server for Linux

To be able to work with Attunity Replicate for Linux and MySQL as a source or target endpoint in a Replicate task, you need to:

- Install MySQL Connector/ODBC for Linux, version 5.2.6 or later, on the Attunity Replicate machine.
- Make sure that the `/etc/odbcinst.ini` file contains an entry for MySQL, as in the following example:

```
[MySQL ODBC 5.2.6 Unicode Driver]
Driver = /usr/lib64/libmyodbc5w.so
UsageCount = 1
```

MySQL Replication

Replication enables data from one MySQL database server (the master) to be copied to one or more MySQL database servers (the slaves).

The Replicate MySQL source endpoint can be configured to replicate data from either a master or a slave.

To replicate changes from a slave (CDC), the binary logging parameter `log_slave_updates` needs to be set to `true (1)`.

Enable Binary Logging

To enable binary logging, the following parameters must be configured in MySQL's `my.ini` (Windows) or `my.cnf` (UNIX) files.

Table 8.8 | Required my.ini/my.cnf Parameters for Binary Logging

Parameter	Value
server_id	Any value from 1. Example: `server_id=1`
log-bin=<path>	Path to the binary log file (without an extension). Example: `log-bin=E:\MySql_Logs\BinLog`
binlog_format	Must be: `binlog_format=row`

Table 8.8 | Required my.ini/my.cnf Parameters for Binary Logging (Cont.)

Parameter	Value
expire_logs_days	To prevent disk space issues, it is strongly recommended not to use the default value (0). Example: `expire_logs_days=5`
binlog_row_image	Must be: `binlog_row_image=full`
binlog_checksum	`NONE` or `CRC32` When enabled, this parameter causes the master to write a checksum for each event in the binary log. The default from MySQL 5.6.6 is `CRC32`. Before that, the default is `NONE`. **Note** Only relevant from MySQL 5.6.
log_slave_updates	When replicating from a MySQL slave database server, this value should be set to true (1). If set to 0 (the default) updates on a slave received from a master during replication are not logged to the slave's binary log. The slave's binary log needs to be enabled for this to have an effect.

Cluster Prerequisites

To be able to replicate clustered (NDB) tables (i.e. by connecting Attunity Replicate to any of the cluster nodes), the following parameters must be configured in MySQL's my.ini (Windows) or my.cnf (UNIX) files.

Note When using Percona as a source, there is no need to perform the procedures described in this section.

Table 8.9 | Required my.ini/my.cnf Parameters for Cluster Replication

Parameter	Value
ndb_log_bin	Must be: `ndb_log_bin=on` This ensures that changes in clustered tables will be logged to the binary log.
ndb_log_update_as_write	Must be: `ndb_log_update_as_write=OFF` This prevents writing UPDATEs as INSERTs in the binary log.

Table 8.9 | Required my.ini/my.cnf Parameters for Cluster Replication (Cont.)

Parameter	Value
ndb_log_updated_only	Must be: `ndb_log_updated_only=OFF` Ensures that the binary log will contain the entire row and not just the changed columns.

Limitations

The following limitations apply:

- When only the LOB column in the source table is updated, Replicate will not update the corresponding target LOB column. The target LOB column will only be updated if at least one other column is also updated in the same transaction.
- Due to the way MySQL operates, when loading data to a MySQL target during a Full Load task, duplicate key errors will not be reported to the logs.
- When updating a column's value to its existing value, a zero rows affected is returned from MySQL (unlike Oracle and SQL Server that perform an update of one row). This generates an entry in the attrep_apply_exceptions control table and the following warning:

  ```
  Some changes from the source database had no impact when applied to the target database. See attrep_apply_exceptions table for details.
  ```

Security Requirements

The Attunity Replicate user must have the ReplicationAdmin role with the following privileges (according to task type):

- REPLICATION CLIENT - Required for Change Processing tasks only. In other words, Full Load only tasks do not require this privilege.
- REPLICATION SLAVE - Required for Change Processing tasks only. In other words, Full Load only tasks do not require this privilege.
- SUPER - Only required in versions *prior* to MySQL 5.6.6.

The Attunity Replicate user must also have SELECT privileges for the source tables designated for replication.

Setting up Amazon RDS MySQL for CDC (Change Data Capture)

To enable CDC with Amazon RDS MySQL, you need to use Amazon RDS MySQL version 5.6 or 5.7.

To set up Amazon RDS MySQL for CDC:

1. Follow the instructions provided by AWS to create a new Parameter Group (see the Binary Logging Format section):

http://docs.aws.amazon.com/AmazonRDS/latest/UserGuide/USER_LogAccess.Concepts.MySQL.html

2. When creating the new Parameter Group, set the following values:
   - `binlog_format=row`
3. Save the new Parameter Group.
4. If you have an existing instance of Amazon RDS MySQL, edit the instance to use the parameters specified in **Step 2** above. Or, if you are provisioning a new instance of Amazon RDS MySQL, reference the new Parameter Group created in **Step 1** above.

MySQL Database Source Data Types

The following table shows the MySQL database source data types that are supported when using Attunity Replicate and the default mapping to Attunity Replicate data types. When replicating to a MySQL target, the source and target data types are the same, apart from the exceptions described in Homogeneous Replication.

For information on how to view the data type that is mapped in the target, see the section for the target database you are using.

For additional information about Attunity Replicate data types, see Replicate Data Types.

Table 8.10 | MySQL Database Source Data Types with Mapping to Attunity Replicate Data Types when the Target is not MySQL

MySQL Source Data Types	Attunity Replicate **Data Types**
INT	INT4
BIGINT	INT8
MEDIUMINT	INT4
TINYINT	INT1
DECIMAL (10)	NUMERIC (10,0)
BINARY	BYTES (1)
BIT	BOOLEAN
BIT (64)	BYTES (8)
BLOB	BYTES (66535)
LONGBLOB	BLOB
MEDIUMBLOB	BLOB
TINYBLOB	BYTES (255)
DATE	DATE
DATETIME	DATETIME

Table 8.10 | MySQL Database Source Data Types with Mapping to Attunity Replicate Data Types when the Target is not MySQL (Cont.)

MySQL Source Data Types	Attunity Replicate Data Types
Note DATETIME without a parenthetical value is replicated without milliseconds, whereas DATETIME with a value of 1-5 - e.g. DATETIME(5) - is replicated with milliseconds.	
Note When replicating a DATETIME column, the time remains the same on the target (i.e. it is not converted to UTC).	
TIME	STRING
TIMESTAMP	DATETIME
Note When replicating a TIMESTAMP column, the time is converted to UTC on the target.	
YEAR	INT2
DOUBLE	REAL8
FLOAT If the FLOAT values are not in the range specified below, use a transformation to map FLOAT to STRING. For an explanation of how to do this, see Using the Transform Tab. Supported FLOAT range: - 1.79E+308 to -2.23E-308, 0 and 2.23E-308 to 1.79E+308	REAL (DOUBLE)
*VARCHAR (45)	WSTRING (45)
*VARCHAR (2000)	WSTRING (2000)
*VARCHAR (4000)	WSTRING (4000)
VARBINARY (4000)	BYTES (4000)
VARBINARY (2000)	BYTES (2000)
*CHAR	WSTRING
*TEXT	WSTRING (65535)

Table 8.10 | MySQL Database Source Data Types with Mapping to Attunity Replicate Data Types when the Target is not MySQL (Cont.)

MySQL Source Data Types	Attunity Replicate Data Types
*LONGTEXT	NCLOB
*MEDIUMTEXT	NCLOB
*TINYTEXT	WSTRING (255)
GEOMETRY	BLOB
POINT	BLOB
LINESTRING	BLOB
POLYGON	BLOB
MULTIPOINT	BLOB
MULTILINESTRING	BLOB
MULTIPOLYGON	BLOB
GEOMETRYCOLLECTION	BLOB
ENUM	WSTRING (Length) Where "Length" is the longest value in the ENUM.
SET	WSTRING (Length) Where "Length" is the total of all values in the SET, including commas.

Note If the DATETIME and TIMESTAMP data types are specified with a "zero" value (i.e. 0000-00-00), you need to make sure that the target database in the replication task supports "zero" values for the DATETIME and TIMESTAMP data types. If they are not supported, you can use a transformation to specify a supported value (e.g. 1970.) Otherwise, they will be recorded as null on the target.

Note The JSON data type introduced in Amazon RDS for MySQL 5.7 is not supported. Consequently, JSON columns in the source tables will be ignored.

Homogeneous Replication

The following section describes how Replicate handles replication between a MySQL source and a MySQL target (i.e. homogeneous replication).

Note In homogeneous replication, the source data first passes through the Attunity Replicate data type and is therefore subject to any limitations of that type.

For information on Replicate data types and their limitations (where relevant), see Replicate Data Types.

For information on which Replicate data types the source data passes through when replicating from MySQL, see the MySQL to Attunity Replicate data types mapping table described earlier.

Data Types

When replicating to a MySQL target endpoint, the data types will be identical with the following exceptions:

MySQL Source Data Types	MySQL Target Data Types
NUMERIC	DECIMAL
LONG VARBINARY	MEDIUMBLOB

Collation

When replicating from one MySQL endpoint to another, table and column collations will be replicated to the target. Collatable data types are indicated by an asterisk (*) in Table 11–3 above.

To support collation replication, the DBA must ensure that the collations defined for the source MySQL database are the same as those defined for the target MySQL database.

Non-Nullable Columns and Primary/Unique Index Names

Non-nullable columns and Primary/Unique Index names are preserved during homogeneous replication.

Setting General Connection Properties

This section describes how to configure general connection properties. For an explanation of how to configure advanced connection properties, see **Setting Advanced Connection Properties** below.

Note You can also use MySQL files as a source. For more information, see Using the Attunity Replicate File Channel.

To add a MySQL source endpoint to Attunity Replicate:

1. In the Attunity Replicate Console, click **Manage Endpoint Connections** to open the **Manage Endpoints Connections** dialog box. Then click the **New Endpoint Connection** button. For more information on adding an endpoint to Attunity Replicate, see Working with Endpoints.
2. In the **Name** field, type a name for your endpoint. This can be any name that will help to identify the database being used.
3. In the **Description** field, type a description that helps to identify the MySQL database. This is optional.
4. Select **SOURCE** as the endpoint **role**.
5. Select **MySQL** as the database **Type**.
6. In the **Server Name** field, enter the host name or IP address of the computer on which the MySQL database is installed.
7. Optionally, change the default port (3306).
8. Type the MySQL authentication information (**User Name**, **Password**) for the authorized user for this MySQL database. If you do not know this information, see your MySQL database Administrator (DBA).

Note Consider the following:

- This information is required. If you are using the **Advanced** tab to create a custom string, make sure to include the **User Name** and **Password** properties. See Setting Advanced Connection Properties for more information.
- This information is case sensitive.
- If you want to set custom properties for this database, see Setting Advanced Connection Properties.

Important: Make sure that the MySQL user entered in the MySQL Authentication section has the correct access privileges. For information on how to provide the required privileges, see Security Requirements.

Selecting a Schema

You can choose which MySQL database to access. After configuring the MySQL source database connection settings, open the **Select Tables** dialog box (by clicking the **Table Selection** button on the right of the console) and select which schema to use from the **Schema** drop down list.

See also Designing Tasks.

Setting Advanced Connection Properties

In the **Advanced** tab, you can set the following parameters:

- **Check binary log for new events every**: Specify how often to check the binary log for changes when the endpoints is idle.
- **Additional ODBC connection properties**: Specify any additional ODBC connection parameters that may be required.

Note Attunity Replicate assumes that MySQL Client 5.2.6 to 5.3.x for Linux or MySQL ODBC Client 5.2.6 to 5.3.x 64-bit for Windows is installed on the Attunity Replicate Server machine. If a version later than 5.3.x is installed, you need to specify the version number as an internal parameter where `provider` is the **Parameter** and `MySQL ODBC <version> Unicode Driver` is the **Value** (where `<version>` is the client version e.g. 5.4).

For instructions on setting internal parameters, see Internal Parameters.

Internal Parameters

Internal parameters are parameters that are not exposed in the UI. You should only use them if instructed by Attunity Support.

To add internal Attunity Replicate parameters:

1. Click the **Internal Parameters** link.

   The **Internal Parameters** dialog box opens.
2. In the edit box, type the name of the parameter you need to add and then click it.
3. The parameter is added to the table below the search box with its default value.
4. Change the default value as required.
5. To reset the parameter value to its default, click the "Restore default value" icon at the end of the row.

Settings Summary

You can view a summary of your settings by clicking the **Setting Summary** link. This is useful if you need to send a summary of your settings to Attunity Support.

Using Hadoop as a Source

This section describes how to set up and use Hadoop as the source endpoint in a replication task.

In this section:

Prerequisites

Before you begin to work with a Hadoop cluster as a data source in Attunity Replicate, make sure that the following prerequisites have been met:

- **General:**
  - The Hadoop WebHDFS must be accessible from the Attunity Replicate machine.
  - The Hadoop Data Nodes must be accessible from the Attunity Replicate machine.
  - The Hadoop WebHDFS service must be running.
  - To access Hive using WebHCat, the Hadoop WebHCat service must be running. Other methods for accessing Hive are described later in this chapter.
  - The user specified in the Attunity Replicate Hadoop target settings must have access to HCatalog.
- **SSL:** Before you can use SSL, you first need to perform the following tasks:
  - Configure each NameNode and each DataNode with an SSL certificate (issued by the same CA).
  - Place the CA certificate on the Replicate Server machine. The certificate should be a base64-encoded PEM (OpenSSL) file.
- **Permissions:** The user specified in the Hadoop source settings must have read permission for the HDFS directories that contain the data files.

Limitations

The following limitations apply:

- Change data capture is not supported. The Hadoop endpoint will not be available for selection if either **Apply Changes** or **Store Changes** is enabled in the task settings.
- Replicating data compressed using a method other than gzip is currently not supported.
- Replicating data that is not in Text format is currently not supported.

- Replicating from tables with skews, buckets or partitions is currently not supported.
- Limited LOB support only.
- Due to a Hive limitation, in Hive version 0.12 and earlier versions, only alphanumeric and underscore characters are allowed in table and column names. From Hive 0.13, column names can contain any Unicode character.

Required Permissions

The Hadoop NameNode must be accessible from the Attunity Replicate machine.

Hadoop Endpoint Source Data Types

The following table shows the Hadoop data types that are supported when using Attunity Replicate and the default mapping to Attunity Replicate data types.

For information on how to view the data type that is mapped in the target, see the section for the target endpoint you are using.

For additional information about Attunity Replicate data types, see Replicate Data Types.

Table 8.11 | Supported Hadoop Data Types Mapped to Attunity Replicate Data Types

Hadoop Data Types	Attunity Replicate Data Types
BOOLEAN	BOOL
BINARY	BLOB
DATE	DATE
TIMESTAMP	DATETIME
TINYINT	INT1
SMALLINT	INT2
INT	INT4
BIGINT	INT8
FLOAT	REAL4
DOUBLE	REAL8
VARCHAR	STRING
CHAR	STRING
STRING	CLOB
DECIMAL	NUMERIC

Unsupported Data Types

The Complex Types listed below are not supported:

- ARRAYS
- MAPS
- STRUCTS
- UNION

Setting General Connection Properties

This section describes how to configure general connection properties. For an explanation of how to configure advanced connection properties, see **Setting Advanced Connection Properties** below.

To add a Hadoop source endpoint to Attunity Replicate:

1. In the Attunity Replicate console, click the Manage Endpoint Connections toolbar button to open the Manage Endpoints Connections dialog box. Then click the **New Endpoint Connection** button. For more information on adding an endpoint to Attunity Replicate, see Working with Endpoints.
2. In the **Name** field, type a name for your endpoint. This can be any name that will help to identify the endpoint being used.
3. In the **Description** field, type a description that helps to identify the Hadoop endpoint. This is optional.
4. Select **SOURCE** as the endpoint **Role**.
5. Select **Hadoop** as the endpoint **Type**.
6. In the **Hadoop NameNode** field, enter the host name or IP address of the Hadoop NameNode machine.

   Note Consider the following:
   - This information is case sensitive.
   - To determine if you are connected to the endpoint you want to use or if the connection information you entered is correct, click **Test Connection**.

     If the connection is successful, a green confirmation message is displayed. If the connection fails, an error message is displayed at the bottom of the dialog box.

     To view the log entry if the connection fails, click **View Log**. The server log is displayed with the information for the connection failure. Note that this button is not available unless the test connection fails.

7. In the **Security section**, do the following:
   a. To encrypt the data between the Replicate machine and HDFS, select **Use SSL**. In order to use SSL, first make sure that the SSL prerequisites described in Prerequisites has been met.

      In the **CA path** field, either specify the directory containing the CA certificate.

      -OR-

      Specify the full path to a specific CA certificate.

b. Select one of the following authentication types:

- **User name** - Select to connect to the Hadoop cluster with only a user name. Then, in the **User name** field, specify the name of a user authorized to access the Hadoop cluster.
- **Kerberos** - Select to authenticate against the Hadoop cluster using Kerberos. Replicate automatically detects whether Attunity Replicate Server is running on Linux or on Windows and displays the appropriate settings.

  Attunity Replicate Server on Linux:

  When Attunity Replicate Server is running on Linux, select either **Ticket** or Keytab from the **Kerberos options** drop-down list.

  If you selected **Ticket**, select one of the following options:

  - **Use global Kerberos ticket file** - Select this option if you want to use the same ticket for several Hadoop endpoints (source or target). In this case, you must make sure to select this option for each Hadoop endpoint instance that you define.
  - **Use specific Kerberos ticket file** - Select this option if you want to use a different ticket file for each Hadoop endpoint (source or target). Then specify the ticket file name in the designated field.

    This option is especially useful if you need to perform a task-level audit of Replicate activity (using a third-party tool) on the Hadoop NameNode. To set this up, define several instances of the same Hadoop endpoint and specify a unique Kerberos ticket file for each instance. Then, for each task, simply select a different Hadoop endpoint instance.

  Note You need to define a global Kerberos ticket file even if you select the **Use specific Kerberos ticket file** option. The global Kerberos ticket file is used for authentication when selecting a Hive endpoint, when testing the connection (using the **Test Connection** button), and when selecting which tables to replicate.

  Note When replicating from a Hadoop source endpoint to a Hadoop target endpoint, both endpoints must be configured to use the same ticket file.

  For additional steps required to complete setup for Kerberos ticket-based authentication, see Using Kerberos Authentication.

  If you selected **Keytab**, provide the following information:

  - **Realm:** The name of the realm in which your Hadoop cluster resides.

    For example, if the full principal name is `john.doe@EXAMPLE.COM`, then `EXAMPLE.COM` is the realm.
  - **Principal:** The user name to use for authentication. The principal must be a

member of the realm entered above.

For example, if the full principal name is `john.doe@EXAMPLE.COM`, then `john.doe` is the principal.

- **Keytab file**: The full path of the Keytab file. The Keytab file should contain the key of the **Principal** specified above.

Attunity Replicate Server on Windows:

When Attunity Replicate Server is running on Windows, select one of the following:

- **Use the following KDC:** Select **Active Directory** (default) if your KDC is Microsoft Active Directory or select **MIT** if your KDC is MIT KDC running on Linux/UNIX.

  Note When the Replicate KDC and the Hadoop KDC are in different domains, a relationship of trust must exist between the two domains.

- **Realm:** The name of the realm/domain in which your Hadoop cluster resides (where *realm* is the MIT term while *domain* is the Active Directory term).
- **Principal:** The username to use for authentication. The principal must be a member of the realm/domain entered above.
- When **Active Directory** is selected - **Password:** The password for the principal entered above.
- When **MIT** is selected - **Keytab file**: The keytab file containing the principal entered above.

Note When replicating from a Hadoop source endpoint to a Hadoop target endpoint, both endpoints must be configured to use the same parameters (KDC, realm, principal, and password).

If you are unsure about any of the above, consult your IT/security administrator.

For additional steps required to complete setup for Kerberos authentication, see Using Kerberos Authentication.

- **User name and password** - Select to connect to the Hadoop NameNode or to the Knox Gateway (when enabled - see below) with a user name and password. Then, in the **User name** and **Password** fields, specify the required user name and password.

  Note Consider the following:
  - A user name and password is required to access the MapR Control System.
  - This information is case sensitive.

Important: Make sure that the specified user has the required Hadoop access privileges. For information on how to provide the required privileges, see Required Permissions.

8. If you need to access the Hortonworks Hadoop distribution through a Knox Gateway, select **Use Knox Gateway**.

   Note To be able to select this option, first select **Use SSL** and then select **Password** from the **Authentication type** drop-down list.

9. Provide values for the following fields:
   - **Knox Gateway host** - The FQDN (Fully Qualified Domain Name) of the Knox Gateway host.
   - **Knox port** - The port number to use to access the host. The default is "8443".
   - **Knox Gateway path** - The context path for the gateway. The default is "gateway".

     Note The port and path values are set in the **gateway-site.xml** file. If you are unsure whether the default values have been changed, contact your IT department.

   - **Cluster name** - The cluster name as configured in Knox. The default is "Default".
10. In the **HDFS section**, select either **WebHDFS** or **HttpFS** as the HDFS access method. If you are accessing MapR, it is recommended to use HttpFS.

    Note When the **Use Knox Gateway** option is selected, the NameNode, **HttpFS Host**, and **Port** fields described below are not relevant (and are therefore hidden).

11. Do one of the following, depending on whether you selected **WebHDFS** or **HttpFS**:

    If you selected **WebHDFS**:

    a. In the **NameNode** field, specify the IP address of the NameNode.

       Note This is the Active node when High Availability is enabled (see below).

    b. Replicate supports replication from an HDFS High Availability cluster. In such a configuration, Replicate communicates with the Active node, but switches to the Standby node in the event of failover. To enable this feature, select the **High Availability** check box. Then, specify the FQDN (Fully Qualified Domain Name) of the Standby NameNode in the **Standby NameNode** field.

    c. In the **Port** field, optionally change the default port (50070).

    If you selected **HttpFS**:

a. In the **HttpFS Host** field, specify the IP address of the HttpFS host.
b. In the **Port** field, optionally change the default port (14000).

12. In the **Hive Access section**, do the following:

> **Note** When the **Use Knox Gateway** option is selected, the Host and **Port** fields described below are not relevant (and are therefore hidden).

- **Access Hive using** field (WebHCat): This value cannot be changed.
- **Host** field: Specify the IP address of the Hive machine.
- **Port** field: Optionally change the default port.
- **Database** field: Specify the name of the Hive target database.

Setting Advanced Connection Properties

In the **Advanced** tab, you can specify source file delimiters and other properties. Note that the source file delimiters only need to be specified if one of the following is true:

- The Hive version is earlier than 0.13
- The SerDe property names used to create the source data files are different from the Hadoop defaults

The default Hadoop property names are as follows:

- field.delim (**Field delimiter** in the UI)
- serialization.null.format (**Null value** in the UI)
- escape.delim (**Escape character** in the UI)
- line.delim (**Record delimiter** in the UI)
- quote.delim (**Quote character** in the UI)

In the **Advanced** tab, you can set the parameters described in the following table.

Table 8.12 | Hadoop Source Endpoint - Advanced Tab Options

Option	Description
Field delimiter	The delimiter used to separate fields in the source files.
Null value	The value used to indicate a null value in the source files. **Example (where @@ is the null value):** mike,male,295678 sara,female,@@
Escape character	This can either be the character used to escape the field delimiter character -

Table 8.12 | Hadoop Source Endpoint - Advanced Tab Options (Cont.)

Option	Description
	if the source files were created using a SerDe that does not support quote characters (see Example 1) or the character used to escape the quote character - if the source files were created using a SerDe that supports quote characters (see Example 2). **Example 1 (where \ is the escape character and a comma is the field delimiter):** sunroof\,power-steering **Example 2 (where \ is the escape character and double quotes is the quote character):** "\"sunroof, power-steering\""
Record delimiter	The delimiter used to separate records (rows) in the source files. If the LazySimpleSerde SerDe was used, the record delimiter must be \n.
Quote character	The character used to escape the field delimiter character in the source files. When a field delimiter is escaped, it is interpreted as actual data, and not as a field delimiter. **Example (where double-quotes is the quote character):** "sunroof,power-steering"

Internal Parameters

Internal parameters are parameters that are not exposed in the UI. You should only use them if instructed by Attunity Support.

To add internal Attunity Replicate parameters:

1. Click the **Internal Parameters** link.
   The **Internal Parameters** dialog box opens.
2. In the edit box, type the name of the parameter you need to add and then click it.
3. The parameter is added to the table below the search box with its default value.
4. Change the default value as required.
5. To reset the parameter value to its default, click the "Restore default value" icon at the end of the row.

Settings Summary

You can view a summary of your settings by clicking the **Setting Summary** link. This is useful if you need to send a summary of your settings to Attunity Support.

Using Teradata Database as a Source

This section describes how to set up and use Teradata Database as a source in a replication task.

In this section:

Prerequisites

The following section describes the prerequisites for working with Attunity Replicate and a Teradata Database Source.

Replicate Server for Windows

Teradata Database ODBC Driver for Windows version 15.00 or above must be installed on the Attunity Replicate Server machine.

Replicate Server for Linux

The following section describes the steps you need to perform to work with Attunity Replicate for Linux and Teradata Database as a source database in a Replicate task. Teradata Database Client requires the DataDirect ODBC driver manager (provided with Teradata Database Client).

1. Install Replicate on the Linux machine as described in Attunity Replicate on Linux: Installing, Upgrading and Uninstalling.
2. Install Teradata Database Client 14.10 or above for Linux.

   Note These instructions assume that the Teradata Database Client is installed in the following location:

   ```
   /opt/teradata/client/14.10
   ```

3. Run the following command:

   ```
   export AREP_ODBC_DRIVER_MANAGER=/opt/teradata/client/14.10/odbc_
   64/lib/libodbc.so
   ```

4. Run the following command:

   ```
   export LD_LIBRARY_PATH=$LD_LIBRARY_
   PATH:/usr/lib64:/opt/teradata/client/14.10/tbuild/lib64:/opt/attunity/rep
   licate/lib
   ```

5. To make sure that the path is updated, run the following command:
   `echo $LD_LIBRARY_PATH`
6. Run the following command:
   `export ODBCINI=/opt/teradata/client/14.10/odbc_64/odbc.ini`
7. Add the export commands to the **site_arep_login.sh file**.
8. Add the Teradata Database name to the hosts file as described in Editing the Hosts File.

Important: A Replicate task cannot be defined with endpoints that use different ODBC Driver Managers. Teradata Database source is accessed using the `DataDirect` ODBC Driver Manager. With the exception of Oracle, Hadoop, File and Replicate Connect sources (which are not subject to the above limitation), all other source endpoints are accessed using the `unixODBC` Driver Manager.

To configure a task with a `DataDirect` source and a `unixODBC` target, you need to use the Replicate File Channel. For more information about setting up a task using the File Channel, see Using the Attunity Replicate File Channel.

Required Permissions

The user that is specified in the **General** tab when Setting General Connection Properties must be registered as a user in the Teradata Database.

Teradata Source Data Types

The following table shows the Teradata target data types that are supported when using Attunity Replicate and the default mapping to the Attunity Replicate data types.

For additional information about Attunity Replicate data types, see Replicate Data Types.

Table 8.13 | Supported Teradata Source Data Types with Mapping to Attunity Replicate Data Types

Teradata Data Types	Attunity Replicate Data Types
BLOB	BLOB
BYTE	BYTES
BYTEINT	INT1
BIGINT	INT8
DATE	DATE
DECIMAL	REAL8
DOUBLE PRECISION	REAL8

Table 8.13 | Supported Teradata Source Data Types with Mapping to Attunity Replicate Data Types (Cont.)

Teradata Data Types	Attunity Replicate Data Types
FLOAT	REAL8
INTEGER	INT4
INTERVAL DAY	STRING (Support a maximum of 9,999 days)
INTERVAL DAY TO HOUR	STRING
INTERVAL DAY TO MINUTE	STRING
INTERVAL DAY TO SECOND	STRING
INTERVAL HOUR	STRING
INTERVAL HOUR TO MINUTE	STRING
INTERVAL HOUR TO SECOND	STRING
INTERVAL MINUTE	STRING
INTERVAL MINUTE TO SECOND	STRING
INTERVAL SECOND	STRING (Supports up to six fractional seconds)
CHAR	STRING
CLOB	CLOB
GRAPHIC	STRING
INTERVAL MONTH	STRING
INTERVAL YEAR	STRING
INTERVAL YEAR TO MONTH	STRING
REAL	REAL8
SMALLINT	INT2
TIME	TIME
TIMESTAMP	DATETIME
TIMESTAMP WITH TIME ZONE	DATETIME
TIME WITH TIME ZONE	TIME
VARBYTE	BYTES
VARCHAR	STRING (10)
VARGRAPHIC	STRING (10)
NUMERIC	NUMERIC
CHAR VARYING	STRING
LONG VARCHAR	STRING

Setting General Connection Properties

This section describes how to configure general connection properties. For an explanation of how to configure change processing parameters, see Configuring Change Processing below.

To add a Teradata Database source endpoint to Attunity Replicate:

1. In the Attunity Replicate console, click **Add Database** to open the Add Endpoints dialog box. For more information on adding an endpoint to Attunity Replicate, see Working with Endpoints.
2. In the **Name** field, type a name for your Teradata database. This can be any name that will help to identify the database being used.
3. In the **Description** field, type a description that helps to identify the Teradata database. This is optional.
4. Select **SOURCE** as the database **role**.
5. Select **Teradata Database** as the database **Type**.
6. Type the **Server** name. This is the name of the computer with the Teradata Database instance you want to work with.
7. Type the Teradata Database authentication information (**Username**, **Password**) for the authorized user for this Teradata Database. If you do not know this information, see your Teradata Database system manager.

   Note Consider the following:

   - This information is case sensitive.
   - To determine if you are connected to the database you want to use or if the connection information you entered is correct, click **Test Connection**.

     If the connection is successful a message in green is displayed. If the connection fails, an error message is displayed at the bottom of the dialog box.

     To view the log entry if the connection fails, click **View Log**. The server log is displayed with the information for the connection failure. Note that this button is not available unless the test connection fails.

   Important: Make sure that the Teradata Database user entered in the Teradata Database Authentication section has the correct access privileges. For information on how to provide the required privileges, see Security Requirements.

8. Type the **Default database** name or select one from the list of available endpoints. This is the name of the Teradata Database where you are replicating the data to.

Setting Change Processing Parameters

The **Change Processing** tab lets you define change processing settings for the Teradata Database source. Normally, Replicate scans a database's transaction logs for changes and

then applies those changes to the target database. However, this method of change processing is not possible with Data Warehouse endpoints such as Teradata Database since these endpoints do not generate transaction logs.

The good news is that you can still use Replicate to capture changes from Teradata Database - it just requires a little bit of preparation.

Prerequisites

Before you can define the settings in the **Change Processing** tab, you need to ensure that at least one special "Context" column exists in your source database tables. Context column(s) are basically columns in a table that enable Replicate to determine whether the data has changed. You can add Context columns specifically for the purpose of change processing (either using a script or manually) or you can use existing columns that contain suitable "Context" data.

Note You can create and reference any number of Context columns in a table as long as the Context column names are the same for all source tables. Additionally, each value in the Context column(s) must be unique.

In the example below, the Context column **cf** has been added to the table. The **cf** column contains TIMESTAMPs that enable Replicate to determine whether a change occurred (by comparing the current TIMESTAMP with the TIMESTAMP stored in its repository).

By default, all changes are assumed to be INSERTs. If UPDATE and DELETE operations are also performed on the source tables, you can write an UPDATE and/or DELETE expression (described below) that will enable Replicate to identify the operation type.

Figure 8.1 | Example of a Table with a Context Column

a	first_name	cf	oper
1	may1	2014-12-29 14:52:2...	D
2	may2	2014-12-29 14:52:2...	D
6	omri6	2014-12-29 14:53:5...	D
8	omri8	2014-12-29 14:53:5...	D
9	omri9	2014-12-29 14:53:5...	U
10	omri10	2014-12-29 14:53:5...	U
11	omri11	2014-12-29 14:53:5...	U
3	pitz	2014-12-30 11:04:0...	U
7	gal	2014-12-30 13:27:1...	U
5	gal	2014-12-30 13:29:2...	U
12	gal	2014-12-30 13:41:5...	U

Limitations

The following limitations apply when Change Processing is enabled for the Teradata Database source:

- The "Start from timestamp" run option is not supported. For more information, see Using Advanced Run Options.
- If one of the Context columns is part of the Primary Key or Unique Index, then UPDATE and DELETE operations are not supported.
- Context columns cannot be LOB columns
- DDLs are not supported
- When inserting a record and then updating the same record, the task error handling settings should be set as follows:
  - Open the ***<Task Name>*** **Settings** dialog box.
  - Select the **Error Handling|Apply Conflicts** tab.
  - Set a task-specific Apply Conflicts policy as described in Error Handling Settings.
  - From the **No record found for applying an update** drop-down list, select **INSERT the missing target record**.

  For more information on error handling, see Error Handling.

Configuring Change Processing Settings

Perform the following steps to configure change processing settings.

To configure change processing settings:

1. Select the **Change Processing** tab in the Teradata Database source.
2. In the **Columns** field, specify the names of the Context columns. The column names are case-sensitive and must be separated by commas.

   Example:

   context1,context2
3. Choose the sorting order of the Context columns as appropriate (**Ascending** or **Descending**). Note that if the order you select is not the same as the actual sorting order, an error will occur.
4. In the **Check for changes every** field, specify how often to check for changes.
5. Enter expressions that Replicate will use to identify UPDATE and DELETE operations. If you do not enter any expressions or if no match is found for an expression, any row whose context is higher (if the sorting order is **Ascending**) or lower (if the sorting order is **Descending**) than the previous context value will be considered an INSERT.

   Note Expressions must be written in the native syntax of the Teradata Database source. All examples in this section are written using PostgresSQL syntax.

- **Update expression** - Enter an expression for identifying UPDATE operations.
  Example (based on Figure "Example of a Table with a Context Column"**):**
  case when **oper='U'** then 1 else 0 end

  Tip: [Selecting the **UPDATE the existing target record** option in the Apply Conflicts tab, eliminates the need to provide an UPDATE expression.

- **Delete expression** - Enter an expression for identifying UPDATE operations.
  Example (based on Figure "Example of a Table with a Context Column"**):**
  case when **oper='D'** then 1 else 0 end

  Important: In addition to the DELETE expression, DELETE operations should be carried out as "Soft" deletes. This means that the row is not actually deleted from the table, but rather, marked as "deleted".

6. Select **Override connection string parameters** to append the connection string with parameters that are not exposed in the UI. As such parameters are normally not required, they should only be used after consulting with Attunity Support.

Using PostgreSQL as a Source

This section describes how to set up and use a PostgreSQL database as a source in a replication task.

In this section:

Source Prerequisites

The following section lists the prerequisites for working with Attunity Replicate and a PostgreSQL database source.

Client Side

- **Attunity Replicate Server for Windows:**
  - The PostgreSQL ODBC Driver `psqlodbc_09_03_0300-x64-1` must be installed on the Attunity Replicate machine.

    Note Make sure that the PostgreSQL ODBC installation folder "bin" (e.g. "C:\Program Files\psqlODBC\0905\bin") is added to the system PATH.

- **Attunity Replicate Server for Linux:**

  On the Attunity Replicate machine:
  - Install `postgresql94-9.4.4-1PGDG.<OS Version>.x86_64.rpm`. This is the package that contains the psql executable.

    For example, `postgresql94-9.4.4-1PGDG.rhel7.x86_64.rpm` is the package required for Red Hat 7.
  - Install the ODBC driver `postgresql94-odbc-09.03.0400-1PGDG.<OS version>.x86_64` or above for Linux, where `<OS version>` is the OS of the Attunity Replicate Server machine.

    For example, `postgresql94-odbc-09.03.0400-1PGDG.rhel7.x86_64` is the client required for Red Hat 7.
  - Make sure that the `/etc/odbcinst.ini` file contains an entry for PostgreSQL, as in the following example:

```
[PostgreSQL]
Description = PostgreSQL ODBC driver
Driver = /usr/pgsql-9.4/lib/psqlodbc.so
Setup = /usr/pgsql-9.4/lib/psqlodbcw.so
Debug = 0
CommLog = 1
UsageCount = 2
```

- Make sure that the **test_decoding** output plugin (found in the **postgresql94-contrib** package) is installed.
- When the **Apply Changes** task option is enabled, the user specified in the PostgreSQL source database's **General** tab must be granted super-user permissions.

Server Side

- The IP address of the Attunity Replicate machine must be added to the `pg_hba.conf` configuration file.
- The following parameters and values must be set in the `postgresql.conf` configuration file.

```
wal_level = logical
max_replication_slots >=1
```

The `max_replication_slots` value should be set according to the number of tasks that you want to run. For example, to run 5 tasks you need to set a minimum of 5 slots. Slots open automatically as soon as a task starts and remain open, even when task is no longer running. Note that open slots need to be manually deleted.

`max_wal_senders >=1`

The `max_wal_senders` parameter sets the number of concurrent tasks that can run.

- The `wal_sender_timeout` parameter terminates replication connections that are inactive longer than the specified number of milliseconds. The default timeout is 60 seconds. To disable the timeout mechanism (optional), set this parameter to zero.

> **Note** By default, the value of the `wal_sender_timeout` parameter is interpreted by the server as milliseconds. To explicitly specify seconds, append an "s" to the value as in the following example:
>
> `wal_sender_timeout=60s`

For more information on the configuration parameters, see:

http://www.postgresql.org/docs/9.4/static/runtime-config-replication.html

Required Permissions

The user specified in the **General** tab when Setting General Connection Properties must be granted the following permissions in the PostgreSQL database:

- **For Full Load replication:** Standard SELECT on the source database
- **For Apply Changes replication:** SUPERUSER

Source Limitations

The following limitations apply when using PostgreSQL as a source:

- The database name cannot include a semi-colon (;).
- A captured table must have a Primary Key. In the event that a table does not have a Primary Key, the result of DELETE and UPDATE record operations will be unpredictable.
- Updating a Primary Key segment is ignored. In such cases, applying such an update will be identified by the target as an update that did not update any rows and will result in a record written to the exceptions table.
- The "Start Process Changes from Timestamp" run option is not supported.
- Replication of the Before Image is not supported.
- Change processing is not supported on Amazon RDS for PostgreSQL.
- Replication of multiple tables with the same name but a different case (e.g. `table1`, `TABLE1` and `Table1`) may cause unpredictable behavior and is therefore not supported.
- Change processing of [CREATE | ALTER | DROP] table DDLs are supported unless they are held in an inner function/procedure body block or in other nested constructs.

  For example, the following change will not be captured:

```
CREATE OR REPLACE FUNCTION attu.create_distributors1() RETURNS void
LANGUAGE plpgsql
AS $$
BEGIN
 create table attu.distributors1(did serial PRIMARY KEY,name varchar(40)
NOT NULL);
END;
$$;
```

- Change processing of TRUNCATE operations is not supported.
- Replication of partitioned tables is not supported. When a partitioned table is detected, the following occurs:
  - The database will report a list of parent and child tables.
  - The table will be created on the target as a regular table with the same properties as the selected tables.
  - If the parent table in the source database has the same Primary Key value as its child tables, a "duplicate key" error will be generated.

Note In order to replicate partitioned tables from a PostgreSQL source to a PostgreSQL target, you first need to manually create the parent and child tables on the target. Then define a separate task to replicate to those tables. In such a case, the task settings should be configured to "Truncate before loading". For more information on the "Truncate before loading" option, see Full Load Settings.

PostgreSQL Source Data Types

The following table shows the PostgreSQL target data types that are supported when using Attunity Replicate and the default mapping to the Attunity Replicate data types.

For additional information about Attunity Replicate data types, see Replicate Data Types.

Table 8.14 | Supported PostgreSQL Source Data Types with Mapping to Attunity Replicate Data Types

PostgreSQL Data Types	Attunity Replicate Data Types
INTEGER	INT4
SMALLINT	INT2
BIGINT	INT8
NUMERIC(p,s) **Note** When replicating to a PostgreSQL target, NUMERIC without precision or scale will be created	If precision is => 0 and =< 38, then: NUMERIC If precision is => 39, then:

Table 8.14 | Supported PostgreSQL Source Data Types with Mapping to Attunity Replicate Data Types (Cont.)

PostgreSQL Data Types	Attunity Replicate Data Types
as NUMERIC (28,6) on the target.	STRING
DECIMAL(p,s)	If precision is => 0 and =< 38, then: NUMERIC If precision is => 39, then: STRING
REAL	REAL4
DOUBLE	REAL8
SMALLSERIAL	INT2
SERIAL	INT4
BIGSERIAL	INT8
MONEY	NUMERIC(38,4) **Note** The MONEY data type is mapped to FLOAT in Microsoft SQL Server.
CHAR	WSTRING (1)
CHAR(n)	WSTRING (n)
VARCHAR(n)	WSTRING (n)
Note VARCHAR without a length (n) is not recognized as a valid data type by target endpoints. Consequently, if a source column data type is set to VARCHAR without an explicit length, Replicate will set a default length of 8000 bytes. You can change the default by setting the following internal parameter to the required length: `unboundedVarcharMaxSize` See also Internal Parameters.	
TEXT	NCLOB
BYTEA	BLOB

Table 8.14 | Supported PostgreSQL Source Data Types with Mapping to Attunity Replicate Data Types (Cont.)

PostgreSQL Data Types	Attunity Replicate Data Types
TIMESTAMP	DATETIME
TIMESTAMP (z)	DATETIME

Note Replicate only supports ISO formatted textual DATE formats (the default). If other formats are used, an error will be generated. You can change the date format in the **postgresql.conf** file or using the `PGDATESTYLE` environment variable. You can also change the date format at database level.

DATE	DATE
TIME	TIME
TIME (z)	TIME
INTERVAL	STRING (128) - 1 YEAR, 2 MONTHS, 3 DAYS, 4 HOURS, 5 MINUTES, 6 SECONDS
BOOLEAN	STRING (5) TRUE|FALSE
ENUM	STRING (64)
CIDR	STRING (50)
INET	STRING (50)
MACADDR	STRING (18)
BIT (n)	STRING (n)
BIT VARYING (n)	STRING (n)
UUID	STRING
TSVECTOR	CLOB
TSQUERY	CLOB
XML	CLOB
POINT	STRING (255) "(x,y)"
LINE	STRING (255) "(x,y,z)"
LSEG	STRING (255) "((x1,y1),(x2,y2))"
BOX	STRING (255) "((x1,y1),(x2,y2))"
PATH	CLOB "((x1,y1),(xn,yn))"
POLYGON	CLOB "((x1,y1),(xn,yn))"

Table 8.14 | Supported PostgreSQL Source Data Types with Mapping to Attunity Replicate Data Types (Cont.)

PostgreSQL Data Types	Attunity Replicate Data Types
CIRCLE	STRING (255) "(x,y),r"
JSON	NCLOB
ARRAY	NCLOB
COMPOSITE	NCLOB
INT4RANGE	STRING (255)
INT8RANGE	STRING (255)
NUMRANGE	STRING (255)
STRRANGE	STRING (255)
CHARACTER VARYING	If length is specified: WSTRING (LENGTH) If no length is specified: WSTRING (8000)

Homogeneous Replication

When replicating from a PostrgreSQL source to a PostrgreSQL target, most of the source and target data types will be identical. The exceptions are listed in the table below.

Additionally, in homogeneous replication, source column and table collations will be replicated to the target as described in Column and Table Collation.

Note In homogeneous replication, the source data first passes through the Attunity Replicate data type and is therefore subject to any limitations of that type.

For information on Replicate data types and their limitations (where relevant), see Replicate Data Types.

For information on which Replicate data types the source data passes through when replicating from PostgreSQL, see the PostgreSQL to Attunity Replicate data types mapping table described earlier.

Data Type Exceptions

When replicating from one PostgreSQL database to another, source and target data types are identical for all supported PostgreSQL versions, with the following exceptions:

Table 8.15 | Data Type Exceptions in Homogeneous Replication

PostgreSQL Source	PostgreSQL Target
ENUM	STRING
COMPOSITE	STRING
NUMERIC	NUMERIC (28,6)

Column and Table Collation

When replicating from one PostgreSQL database to another, column and table collations will be replicated to the target.

Note To support collation replication, the DBA must ensure that the collations defined for the source PostgreSQL database are the same as those defined for the target PostgreSQL database.

Non-Nullable Columns and Primary/Unique Index Names

Non-nullable columns and Primary/Unique Index names are preserved during homogeneous replication.

Setting General Connection Properties

This section describes how to configure general connection properties. For an explanation of how to configure advanced connection properties, see **Setting Advanced Connection Properties** below.

To add a PostgreSQL endpoint source database to Attunity Replicate:

1. In the Attunity Replicate console, click **Add database** to open the **Add Endpoints** dialog box. For more information on adding an endpoint to Attunity Replicate, see Working with Endpoints.
2. In the **Name** field, type a name for your PostgreSQL database. This can be any name that will help to identify the database being used.
3. In the **Description** field, type a description that helps to identify the PostgreSQL database. This is optional.
4. Select **SOURCE** as the database **role**.
5. Select **PostgreSQL** as the database **Type**.
6. Type the **Server** name. This is the name or IP address of the computer with the PostgreSQL database that you want to access.
7. Optionally, change the default port (5432).
8. Enter the PostgreSQL database authentication information (**User name**, **Password**) of an authorized PostgreSQL user. If you do not know this information, see your

PostgreSQL database system manager.

Note Consider the following:

- This information is case sensitive.
- To determine if you are connected to the database you want to use or if the connection information you entered is correct, click **Test Connection**.

  If the connection is successful a message in green is displayed. If the connection fails, an error message is displayed at the bottom of the dialog box.

  To view the log entry if the connection fails, click **View Log**. The server log is displayed with the information for the connection failure. Note that this button is not available unless the test connection fails.

Important: Make sure that the PostgreSQL database user entered in the PostgreSQL database Authentication section has the correct access privileges.

9. Type the **database name** or select one from the list of available endpoints. This is the name of the PostgreSQL database from which you are replicating data.
10. Click **OK** to save your settings and close the dialog box.

Setting Advanced Connection Properties

In the **Advanced** tab, you can set the following properties:

- **Capture DDLs:** When this option is selected, the following actions occur:
  - Operational artifacts are created (by Replicate) in the database when the task starts. In order to capture DDL events, Attunity Replicate creates various artifacts in the PostgreSQL database when the task starts. You can later remove these artifacts as described in Removing Replicate Artifacts from the PostgreSQL Database.
  - Streamed DDL events are captured.
- **Create DDL artifacts in schema:** The schema in which the operational DDL database artifacts will be created. The default value is "Public".
- **WAL heartbeat** - An Apply Changes task that is running but not capturing changes (due to source table inactivity) will continue to occupy the LSN position in its replication slot, thereby preventing truncation of the WAL. Since the WAL is a server-wide resource used by all PostgreSQL processes, it may grow extremely large if no changes are captured for an extended period.

  To prevent this from happening, enable the "WAL heartbeat" option. When this option is enabled, the PostgreSQL source endpoint mimics task activity by periodically committing pseudo transactions (i.e. "Heartbeats") to the heartbeat table, thereby advancing the task slot's LSN position.
  - **Create WAL heartbeat table in schema:** The schema in which the WAL heartbeat table (`attrep_wal_heartbeat`) will be created. The default value is "public".

- **Heartbeat frequency (minutes):** The frequency with which to commit transactions to the heartbeat table.

Internal Parameters

Internal parameters are parameters that are not exposed in the UI. You should only use them if instructed by Attunity Support.

To add internal Attunity Replicate parameters:

1. Click the **Internal Parameters** link.
   The **Internal Parameters** dialog box opens.
2. In the edit box, type the name of the parameter you need to add and then click it.
3. The parameter is added to the table below the search box with its default value.
4. Change the default value as required.
5. To reset the parameter value to its default, click the "Restore default value" icon at the end of the row.

Settings Summary

You can view a summary of your settings by clicking the **Setting Summary** link. This is useful if you need to send a summary of your settings to Attunity Support.

Removing Replicate Artifacts from the PostgreSQL Database

In order to capture DDLs, Attunity Replicate creates various artifacts in the PostgreSQL database when the task starts. When the task completes, you may wish to remove these artifacts.

To remove the artifacts, issue the following statements (in the order they appear below), where `public` is the default schema in which the artifacts were created:

```
drop event trigger attrep_intercept_ddl;
```

Note that the `event trigger` does not belong to a specific schema.

```
drop function public.attrep_intercept_ddl()

drop table public.attrep_ddl_audit

drop schema public
```

Important: Dropping a schema should be done with extreme caution, if at all. Never drop an operational schema, especially not `public`.

Using a File as a Source

This section describes how to set up and use delimited text files as a source in a replication task. You can use the File target endpoint to export database tables to files, which can then be used as a source in a Replicate task with a File source endpoint.

In this section:

General Overview

The Replicate File endpoint can be used either as a source or as a target. When used as a source, the File endpoint requires the source files to be in delimited text file format. When used as a target, the File The Amazon S3 endpoint generates the data files ether in delimited text file format (CSV) or in JSON format (according to the format selected in the endpoint settings).

Delimited text files are used to store data in tabular format. Examples of delimited text file formats include the CSV (Comma Separated Values) and TSV (Tab Separated Values) formats. Some organizations may implement procedures that export data from a database to a delimited text file while others may simply prefer this format as a convenient way of storing tabular data.

In a delimited text file, each record in the table occupies a separate row. Delimiters are used to mark the beginning of a new row or the beginning of a new column. Virtually any character can be used as a delimiter, although a newline (\n) is often used to separate rows, and commas are commonly used to separate columns.

In JSON files, each record is represented by a single line.

So, for example, the following table:

book_id	title	price	is_hardcover
123	Angels	6.99	false
456	The Fallen	6.49	true
789	Rise Up	7.23	true

Will be represented as:

```
{ "book_id": 123, "title": "Angels", "price": 6.99, "is_hardcover": false }
```

```
{ "book_id": 456, "title": "Fallen", "price": 6.49, "is_hardcover": true }
{ "book_id": 789, "title": "Rise Up", "price": 7.23, "is_hardcover": true }
```

See also File Source Overview and File Target Overview.

File Source Overview

When you configure the File Source endpoint in Replicate, you need to specify which row and column delimiters are used in your source files as well as which characters are used to enclose columns containing delimiters. You may also need to specify which character is used to escape columns enclosed in double-quotes (should such columns exist in your source files). If you have previously used Replicate to transfer endpoint tables to files and you now wish to use those files as a source in another task, then you should specify the same delimiters (i.e. that were used to generate target files).

For more information, see Setting General Properties.

Three types of delimited files are used in the Replicate Source File endpoint:

- Full Load Files
- Change Files
- Reference Files

Reference Files

Change Files can either reside in a single location or in multiple locations. To access Change Files that reside in different locations and/or that do not include the target table names, you need to use a Reference File. If the rows in the Change File(s) contain the names of the tables to which to apply the changes, then the Reference File only needs to contain the paths to the Change Files. If each Change File contains changes for a single table, then each of the Change File paths in the Reference File needs to be preceded by the name of its corresponding target table.

For more information on Reference File and Reference File formats, see Change Processing.

Each row in the Reference File should be formatted as follows:

```
[<table_name>],<full path to Change File>
```

Where `[<table_name>]` is required only if the referenced Change File contains changes for a single table.

Note Reference File names cannot exceed 70 characters (no such limitation exists for the path length). Reference File names that exceed 70 characters will be ignored and appropriate warning will be written to the log.

Example 8.1 | Reference File: Each Change File contains changes for a single table:

```
table1,c:\temp\cdc1.csv
```

```
table2,c:\temp\cdc2.csv
table3,c:\temp\cdc3.csv
```

Example 8.2 | Reference File: Each Change File contains changes for multiple tables:

```
c:\temp\cdc1.csv
c:\temp\cdc2.csv
c:\temp\cdc3.csv
```

Full Load Files

Full Load Files are used to populate the empty source tables with data. Full Load Files should only contain the table data. The source tables themselves are created using the External Table Editor provided in the File source endpoint configuration. Both the tables and their data are replicated to the target endpoint during the Full Load stage of the Replicate task.

Example of a Full Load Data File

```
22,January,2014,male,5463565
12,May,2011,female,3236776
9,March,2009,male,9648675
```

For more information on Full Load data files and creating tables, see Defining Tables and Full Load Data.

Change Files

A Change File is a delimited text file that contains a record of DML changes - represented as rows - to apply to the specified target tables. Replicate reads the Change File(s) and applies the changes to the relevant target tables, which can either be specified in the Change File itself or in a Reference File (see Reference Files below for details). Change Files are picked up from the source directory according to their modification date, thereby ensuring that the changes will be processed in the proper sequence.

Note The Change File modification date *must* be both newer than the task start timestamp and newer than the last processed Change File.

Each row in a Change File consists of the following delimited columns:

- (Optional) The change operation e.g. DELETE. If the operation field is absent, INSERT is assumed.
- The name of the target table to which to apply the change (only required if the Change File contains changes for multiple tables)
- (Optional) The timestamp of the change i.e. when the change occurred
- (Optional) The user who applied the change
- The data to change (one or more columns)

Change Files can either contain changes for multiple tables or for a single table, as shown in the examples below.

Note To access Change Files that reside in different locations and/or that do not include the target table names, you need to use a Reference File. For more information on Reference Files, see Reference Files.

Note Change File names cannot exceed 70 characters (no such limitation exists for the path length). Change File names that exceed 70 characters will be ignored and appropriate warning will be written to the log.

Example 8.3 | Change File that contains changes for multiple tables

```
INSERT,table1,ts1,user,dog,cat,bird
INSERT,table2,ts1,user,dog,cat,bird
DELETE,table3,ts1,user,dog,cat,bird
```

Example 8.4 | Change File that contains changes for a single table

```
INSERT,,ts1,user,dog,cat,bird
INSERT,,ts1,user,dog,cat,bird
DELETE,,ts1,user,dog,cat,bird
```

Prerequisites

Before you begin to work with a File as a source in Attunity Replicate, make sure that the following prerequisites have been met:

- Attunity Replicate installed in your network
- Change Files, Full Load files and Reference Files should be in delimited text file format
- Source files (including the Reference File) should be accessible from the Attunity Replicate machine.

Limitations

The following limitations apply to the File source:

- Change Files that are currently being used in a Replicate task cannot be modified while the task is in progress.
- Stopping a Full Load task and then starting it again will start the task from the beginning (and not from the point at which it was stopped).

Setting General Properties

You can add a File endpoint to Attunity Replicate to use as a source.

For more information on working with endpoints, see Working with Endpoints.

To set general properties for a File source endpoint:

1. In **Tasks** view, click **Manage Endpoint Connections** to open the **Manage Endpoints Connections** dialog box. Then click the **New Endpoint Connection** button.
2. In the **Name** field, type a name for your endpoint. This can be any name that will help to identify the endpoint being used.
3. In the **Description** field, type a description that helps to identify the File endpoint. This is optional.
4. Select **SOURCE** as the endpoint **role**.
5. Select **File** as the endpoint **Type**.
6. Configure the settings in the **General** tab as described in the table below.

Table 8.16 | File Source Endpoint - General Tab Options

Option	Description
File Format	
Field Delimiter	The delimiter used to separate columns in the source files. The default is a comma. **Example:** mike,male
Record delimiter	The delimiter used to separate records (rows) in the source files. The default is a carriage return (`\n`). **Example (Using an asterisk as the row delimiter)** mike,male*sara,female
Null value	The character used to indicate a null value in the source files. **Example (where * is the row delimiter and @ is the null value):** mike,male,295678*sara,female,@
Quote character	The character used at the beginning and end of a column that contains the column delimiter character. The default is the double-quote character ("). When a column that contains column delimiters is enclosed in double-quotes, the column delimiter characters are interpreted as actual data, and not as column delimiters. **Example (where a comma is the column delimiter):** "sunroof, power-steering"
Escape character	The character used to escape a string when both the string and the column containing the string are enclosed in quotation marks. Note that the string's quotation marks will be removed unless they are escaped.

Table 8.16 | File Source Endpoint - General Tab Options (Cont.)

Option	Description
	Example (where " is the quote character and \ is the escape character): 1955,"old, \"rare\", Chevrolet",$1000
Code page	Specify the code page of your source files if it is different from the default (65001). **Note** Windows and Linux systems use different code page conventions. The specified code page must comply with the code page convention of the source file system.
Ignore records	Optionally, specify which header and footer rows in the source files to ignore. Make sure that the header and footer rows to ignore do not contain actual data.
Change Processing	
Important: Changes cannot be captured from Change Files that are present *during* the Full Load operation. Consequently, the Change Files should be placed in their source location(s) only *after* Full Load completes.	
Folder	Select this option if your Change Files contain the target table names and reside in a single folder. Then specify the folder location in the designated field. You can optionally use wildcard characters to only process files that match the specified pattern. **Example:** `c:\temp\*changes.CSV` See also: Change Files.
Use Reference Files	Select this option if you are using Reference Files to point to the location of the Change Files. Then specify one of the following: » The path to the Reference Files » The path to a specific Reference File » A path with a wildcard pattern that matches the Reference File names (e.g. C:\Reference Files*.csv) The folder can either contain a single reference file (which is continually appended with the Change File locations) or multiple reference files. For information on when Reference Files should be used, see Reference Files.

Table 8.16 | File Source Endpoint - General Tab Options (Cont.)

Option	Description
Change File path is preceded by table name	Select this option if each of the Change File paths in the reference files is preceded by a table name. **Note** Selecting this option will disable the **Table name** check box in the **Header columns ordinal position** section.
Change File Column Order	Specify the position of each column in your Change Files. Apart from the data columns which must be positioned after the other columns, the columns can be positioned in any order. So, for example, if your Change Files looked liked this: `DELETE,table1,timestamp,user1,dog,cat,bird` Then, the column positions would be set as follows: **Operation:** 1 **Table name:** 2 **Timestamp:** 3 **User name:** 4 **Data start:** 5 For more information on the columns, see Change Files.

Note To determine if you are connected to the endpoint you want to use or if the connection information you entered is correct, click **Test Connection**.

If the connection is successful a message in green is displayed. If the connection fails, an error message is displayed at the bottom of the dialog box.

To view the log entry if the connection fails, click **View Log**. The server log is displayed with the information for the connection failure. Note that this button is not available unless the test connection fails.

Defining Tables and Full Load Data

In the **Tables** tab, you specify the location of your Full Load data files and define how the source tables will be created. During Full Load, Replicate will copy the source tables to the target endpoint.

To define a table:

1. In the **Tables** tab, click **New Table**.
   The **New Source Table** window opens.
2. In the **Table name** field, specify the name of the table.

3. In the **Location of full load data file(s)** field, specify location of the delimited text files that contain your full load data. Wildcard characters are supported.
4. To add a field, click **Add Field**. Then, click the default column name to edit it.
5. To specify a data type, select the row and then select a type from the drop-down list in the **Type** column.
6. (Optional) Click in the **Key** column to add or remove a Primary Key to/from a column.
7. To create the table, click **OK**.

   The table will be added to the list of tables in the **Tables** tab.

   See also the example for Creating a Table.

To edit a field:

- Double-click the column in the **New/Edit Source Table** dialog box and then edit the values as described above.

To delete a field:

- Select the column in the **New/Edit Source Table** dialog box and then click the **Delete Field** button.

To change the position of a field:

- Select the field and then click the **Up**/**Down** and **Move to Top**/**Move to Bottom** buttons as required.

To edit a table:

- In the **Tables** tab, either double-click the table or select the table and then click the **Edit** button. Edit the table as described above.

To delete a table:

- In the **Tables** tab, select the table and then click the **Delete** button.

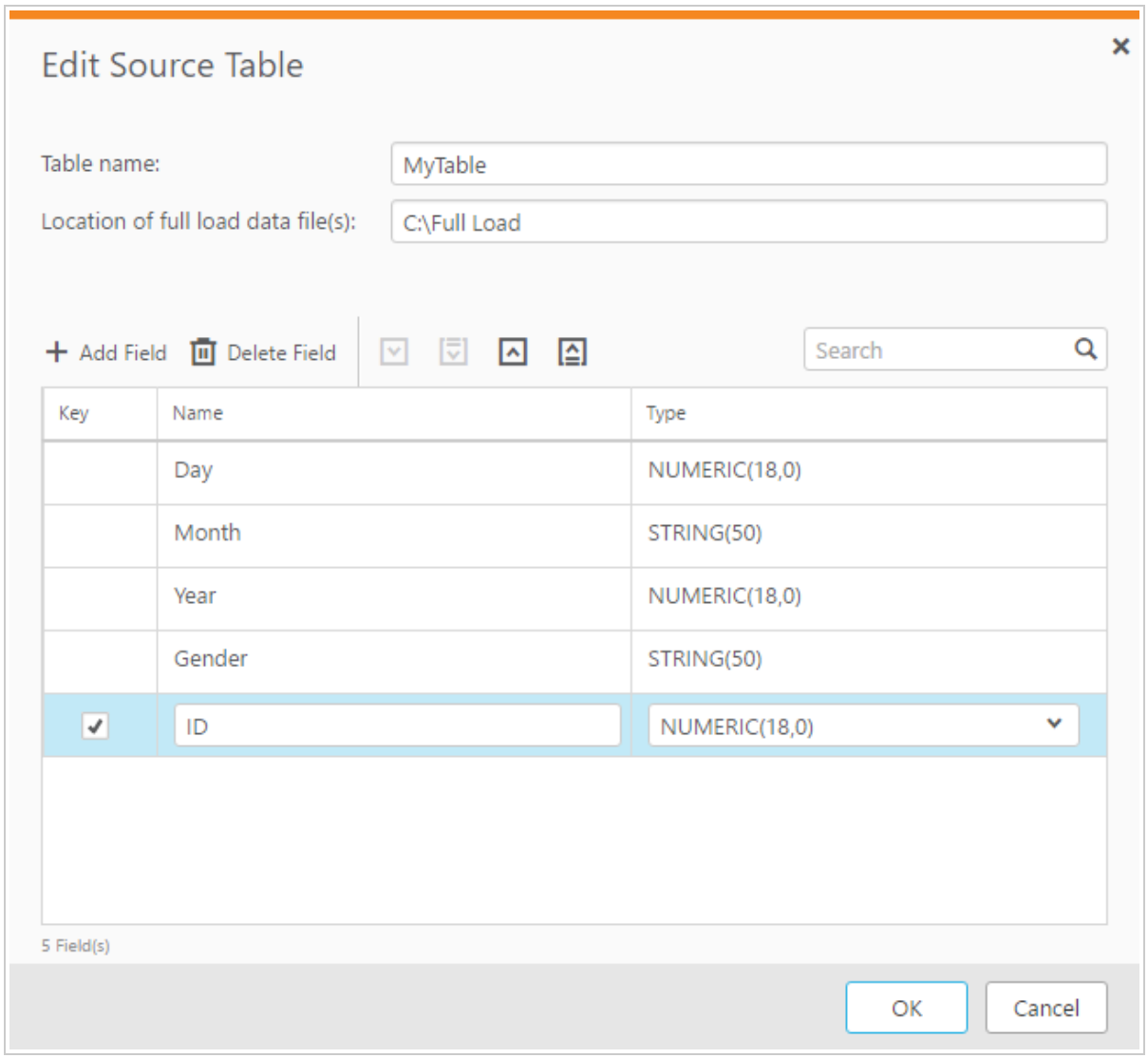
Creating a Table

The source table definition must match the column data values in the Full Load file(s). So, for example, if the Full Load data file contains the following delimited columns:

```
22,January,2014,male,5463565
12,May,2011,female,3236776
9,March,2009,male,9648675
30,June,2002,female,3458795
```

Note Boolean values must be expressed as the digits 1 (TRUE) or 0 (FALSE) and not TRUE or FALSE.

Then the table definitions would look something like this:



Setting Advanced Properties

In the **Advanced** tab, the following options are available.

Table 8.17 | File Source Endpoint - Advanced Tab Options

Option	Description
File pre-processing command	If your source files (Full Load and/or Change Files) are not in delimited text format, you can convert them to the required format using a conversion program. The command should be specified as in the following example: `c:\temp\files\convertfile.exe`

Table 8.17 | File Source Endpoint - Advanced Tab Options (Cont.)

Option	Description
	Note The path is only necessary if the conversion program's location is not defined in the "Path" system variable. The conversion program should accept the following parameters: » The location of the input file(s) (as specified in the **Change Processing** settings) » The output file The output file will be written to the following location: `PRODUCT_INSTALLATION\prod\data\tasks\TASK_NAME\trans_files\`
Check for changes every	Specify how often to check the Change Files for updates.
Change Processing Cleanup	Select one of the following cleanup options to determine what Replicate should do with the processed Change Files/Reference Files: » **Do nothing** - to leave the file(s) in the original location. » **Delete files** - to delete the file(s) from the disk. » **Archive files to folder** - to archive the file(s) to the specified location. **Note** In the case of Reference Files, the **Delete files** and **Archive files to folder** operations will only be performed if there are multiple reference files.

Internal Parameters

Internal parameters are parameters that are not exposed in the UI. You should only use them if instructed by Attunity Support.

To add internal Attunity Replicate parameters:

1. Click the **Internal Parameters** link.
   The **Internal Parameters** dialog box opens.
2. In the edit box, type the name of the parameter you need to add and then click it.
3. The parameter is added to the table below the search box with its default value.
4. Change the default value as required.
5. To reset the parameter value to its default, click the "Restore default value" icon at the end of the row.

Settings Summary

You can view a summary of your settings by clicking the **Setting Summary** link. This is useful if you need to send a summary of your settings to Attunity Support.

Using ODBC with CDC as a Source

This section describes how to use ODBC connectivity to connect to a source endpoint in a Full Load and/or CDC task.

In this section:

Prerequisites

The following section describes the prerequisites for working with Attunity Replicate and an ODBC source with CDC.

Replicate Server for Windows

You can connect an endpoint to Attunity Replicate using ODBC by indicating the DSN (Data Source Name). In this case you must be sure that a DSN is defined for the ODBC endpoint on the computer where Attunity Replicate is installed.

1. Install an endpoint client on the computer where Attunity Replicate is installed. The client you install depends on the ODBC provider you are using. For example, if you are using an IBM DB2 endpoint, install an IBM DB2 client.

   Note You must use a 64-bit ODBC provider client to work with Attunity Replicate

2. Use the ODBC Data Source Administrator to create a System DSN.The Data Source is located in the Windows control panel.

Replicate Server for Linux

The following section describes the steps for working with Attunity Replicate for Linux and ODBC with CDC as a source endpoint in a Replicate task.

1. On the Attunity Replicate Server machine, install the ODBC client that you want to use (e.g. postgreSQL).
2. Makes sure that the `/etc/odbcinst.ini` file contains the correct entry for the driver you installed, as in the following example: `[PostgeSQL]`

```
Description = ODBC for PostgreSQL
Driver = /usr/lib/psqlodbc.so
Setup = /usr/lib/libodbcpsqlS.so
Driver64 = /usr/lib64/psqlodbc.so
```

```
Setup64 = /usr/lib64/libodbcpsqlS.so
FileUsage = 1
```

3. Define a DSN for the installed driver by editing the `/etc/odbc.ini` file, as in the following example:

```
[Postgre_DSN]
Description = Test
Driver = /usr/lib64/psqlodbc.so
Endpoint = MyDatabase
Servername = 12.3.45.678
Port = 5432
```

Limitations

The following limitations apply:

- UPDATES to primary key fields are not supported. To update the field, define it as a unique index instead.
- For providers that do not support batch operations, you must manually add the `RowByRow=true` internal parameter according to the instruction provided in Configuring Change Processing Settings.
- The "Resume from timestamp" run option is not supported.

ODBC with CDC Source Data Types

The following table shows the ODBC target data types that are supported when using Attunity Replicate and the default mapping from Attunity Replicate data types.

For information on how to view the data type that is mapped in the target, see the section for the target endpoint you are using.

For additional information about Attunity Replicate data types, see Replicate Data Types.

Table 8.18 | Supported ODBC with CDC Source Data Types with Mapping to Attunity Replicate Data Types

ODBC Data Types	Attunity Replicate Data Types
SQL_BIT	**BOOLEAN**
SQL_TINYINT	**INT1** **UINT1** **Note** SQL data types are mapped to unsigned data types when the `UNSIGNED_ATTRIBUTE` is set to `SQL_TRUE` for the data type being mapped.

Table 8.18 | Supported ODBC with CDC Source Data Types with Mapping to Attunity Replicate Data Types (Cont.)

ODBC Data Types	Attunity Replicate Data Types
SQL_SMALLINT	**INT2** **UINT2** **Note** SQL data types are mapped to unsigned data types when the `UNSIGNED_ATTRIBUTE` is set to `SQL_TRUE` for the data type being mapped.
SQL_INTEGER	**INT4** **UINT4** **Note** SQL data types are mapped to unsigned data types when the `UNSIGNED_ATTRIBUTE` is set to `SQL_TRUE` for the data type being mapped.
SQL_BIGINT	**INT8** **UINT8** **Note** SQL data types are mapped to unsigned data types when the `UNSIGNED_ATTRIBUTE` is set to `SQL_TRUE` for the data type being mapped.
SQL_DOUBLE	**REAL8**
SQL_FLOAT	**REAL8**
SQL_REAL	**REAL8**
SQL_NUMERIC (P,S)	**NUMERIC (P,S)** **REAL8** The SQL_NUMERIC data type is mapped to REAL8 when *at least one* of the following is true: » Precision > 38 » Scale < 0 » Scale > 38 » Scale > Precision
SQL_DECIMAL (P,S)	**NUMERIC (P,S)** **REAL 8** The SQL_NUMERIC data type is mapped to REAL8 when *at least one* of the following is true:

Table 8.18 | Supported ODBC with CDC Source Data Types with Mapping to Attunity Replicate Data Types (Cont.)

ODBC Data Types	Attunity Replicate Data Types
	» Precision > 38 » Scale < 0 » Scale > 38 » Scale > Precision
SQL_DATE SQL_TYPE_DATE	**DATE**
SQL_TIME SQL_TYPE_TIME	**TIME**
SQL_TIMESTAMP SQL_TYPE_TIMESTAMP	**DATETIME**
SQL_CHAR SQL_VARCHAR	**STRING**
SQL_WCHAR SQL_WVARCHAR	**WSTRING**
SQL_LONGVARCHAR To use this data type with Attunity Replicate, you must enable the use of CLOBs for a specific task. During CDC, CLOB data types are supported only in tables that include a primary key. For more information, see **LOB support** in **Task Settings**/Metadata.	**CLOB**
SQL_WLONGVARCHAR To use this data type with Attunity Replicate, you must enable the use of NCLOBs for a specific task. During CDC, NCLOB data types are supported only in tables that include a primary key. For more information, see **LOB support** in **Task Settings**/Metadata.	**NCLOB**
SQL_BINARY	**BYTES**
SQL_LONGVARBINARY	**BLOB**

Table 8.18 | Supported ODBC with CDC Source Data Types with Mapping to Attunity Replicate Data Types (Cont.)

ODBC Data Types	Attunity Replicate Data Types
To use this data type with Attunity Replicate, you must enable the use of BLOBs for a specific task. BLOB data types are supported only in tables that include a primary key. For more information, see **LOB support** in **Task Settings**/Metadata.	
SQL_GUID	**STRING**
SQL_INTERVAL_YEAR SQL_INTERVAL_MONTH SQL_INTERVAL_DAY SQL_INTERVAL_MINUTE SQL_INTERVAL_HOUR SQL_INTERVAL_SECOND SQL_INTERVAL_YEAR_TO_MONTH SQL_INTERVAL_DAY_TO_HOUR SQL_INTERVAL_DAY_TO_MINUTE SQL_INTERVAL_DAY_TO_SECOND SQL_INTERVAL_HOUR_TO_MINUTE SQL_INTERVAL_HOUR_TO_SECOND SQL_INTERVAL_MINUTE_TO_SECOND	**STRING**
Provider specific data types If column length is 0 or > 4000 then: To use this data type with Attunity Replicate, you must enable the use of BLOBs for a specific task. BLOB data types are supported only in tables that include a primary key. For more information, see **LOB support** in **Task Settings**/Metadata.	If column length is < or = 4000: **BYTES** If column length is 0 or > 4000: **BLOB**

Setting General Connection Properties

This section describes how to configure general connection properties.

To add an ODBC with CDC source endpoint to Attunity Replicate:

1. In the Attunity Replicate console, click the Manage Endpoint Connections toolbar button to open the Manage Endpoints Connections dialog box. Then click the **New Endpoint Connection** button. For more information on adding an endpoint to Attunity Replicate, see Working with Endpoints.
2. In the **Name** field, type a name for your ODBC endpoint. This can be any name that will help to identify the endpoint being used.
3. In the **Description** field, type a description that helps to identify the ODBC endpoint. This is optional.
4. Select **SOURCE** as the endpoint **role**.
5. Select **ODBC with CDC** as the endpoint **Type**.
6. Select *one* of the following:
   - **DSN**: Select this to connect to an ODBC-supported endpoint using a DSN. When you select DSN you must select the DSN you are using from the list.

     If the DSN you want to use is not included in the list, make sure that the endpoint client is installed on the computer with Attunity Replicate and that the DSN is defined. Note that the ODBC provider client must be 64-bit. For more information, see Prerequisites .

     Note If you are using a Replicate Connect CDC Agent as the source in a Replicate task, you cannot select the DSN for the Attunity ODBC driver as the target. In this case, to use Attunity ODBC as a source, you must enter the connection string manually by selecting **Connection String** and following the directions for that option in this procedure.

   - **Connection String**: Select this to connect to an ODBC-supported endpoint using a connection string then type a valid connection string in the field below. For information on how to create a connection string, see the documentation for the ODBC endpoint provider you are using.

     Note that if you specify a password in your connection string, it will be revealed as plain text in the task log files. It is therefore recommended to specify the password in the GUI **Password** field.

     Note To determine if you are connected to the endpoint you want to use or if the connection information you entered is correct, click **Test Connection**.

     If the connection is successful a message in green is displayed. If the connection fails, an error message is displayed at the bottom of the dialog box.

     To view the log entry if the connection fails, click **View Log** (this button is not available unless the test connection fails). The server log is displayed with the information for the connection failure.

7. Type the authentication information (**User Name**, **Password**) for the authorized user

for the ODBC endpoint being used. For example, the IBM DB2 system administrator if you are using a IBM DB2 provider. If you do not know this information, see your ODBC Endpoint System Administrator.

Note Consider the following:

- If you select **Connection String**, be sure to include **User name/password** information in the connection string that you type in the box.
- This information is case sensitive.

Important: Make sure that the ODBC endpoint user has the correct access privileges for the ODBC provider being used.

Configuring Change Processing

The **Change Processing** tab lets you define change processing settings for the ODBC with CDC source. Normally, Replicate scans an endpoint's transaction logs for changes and then applies those changes to the target endpoint. However, this method of change processing is not possible with ODBC-based endpoints since these endpoints do not generate transaction logs.

The good news is that you can still use Replicate to capture changes from generic ODBC-derived endpoints - it just requires a little bit of preparation.

Prerequisites

Before you can define the settings in the **Change Processing** tab, you need to ensure that at least one special "Context" column exists in your source endpoint tables. Context column (s) are basically columns in a table that enable Replicate to determine whether the data has changed. You can add Context columns specifically for the purpose of change processing (either using a script or manually) or you can use existing columns that contain suitable "Context" data.

Note You can create and reference any number of Context columns in a table as long as the Context column names are the same for all source tables. Additionally, each value in the Context column(s) must be unique.

In the example below, the Context column **cf** has been added to the table. The **cf** column contains TIMESTAMPs that enable Replicate to determine whether a change occurred (by comparing the current TIMESTAMP with the TIMESTAMP stored in its repository).

By default, all changes are assumed to be INSERTs. If UPDATE and DELETE operations are also performed on the source tables, you can write an UPDATE and/or DELETE expression (described in Configuring Change Processing Settings below) that will enable Replicate to identify the operation type.

Figure 8.2 | Example of a Table with a Context Column

a	first_name	cf	oper
1	may1	2014-12-29 14:52:2...	D
2	may2	2014-12-29 14:52:2...	D
6	omri6	2014-12-29 14:53:5...	D
8	omri8	2014-12-29 14:53:5...	D
9	omri9	2014-12-29 14:53:5...	U
10	omri10	2014-12-29 14:53:5...	U
11	omri11	2014-12-29 14:53:5...	U
3	pitz	2014-12-30 11:04:0...	U
7	gal	2014-12-30 13:27:1...	U
5	gal	2014-12-30 13:29:2...	U
12	gal	2014-12-30 13:41:5...	U

If you need to capture UPDATE and DELETE operations, the following additional rules apply:

- The source table/view must have Primary Key or Unique Index.
- The Primary Key or Unique Index must not be changed. If they *are* changed, UPDATE and DELETE operations on the target might behave unpredictably or not happen at all.

Limitations

The following limitations apply when Change Processing is enabled for the ODBC with CDC source:

- The "Start from timestamp" run option is not supported. For more information, see Using Advanced Run Options.
- If one of the Context columns is part of the Primary Key or Unique Index, then UPDATE and DELETE operations are not supported.
- Context columns cannot be LOB columns
- DDLs are not supported
- When inserting a record and then updating the same record, the task error handing settings should be set as follows:
  - Open the ***<Task Name>* Settings** dialog box.
  - Select the **Error Handling|Apply Conflicts** tab.
  - Set a task-specific Apply Conflicts policy as described in Error Handling Settings.
  - From the **No record found for applying an update** drop-down list, select **INSERT the missing target record**.

  For more information on error handling, see Error Handling.

Configuring Change Processing Settings

1. Select the **Change Processing** tab in the "ODBC with CDC" source endpoint.
2. In the **Columns** field, specify the names of the Context columns. The column names are case-sensitive and must be separated by commas.

   Example:

   ```
   cf1,cf2
   ```
3. Choose the sorting order of the Context columns as appropriate (**Ascending** or **Descending**). Note that if the order you select is not the same as the actual sorting order, an error will occur.
4. In the **Check for changes every** field, specify how often to check for changes.
5. Enter expressions that Replicate will use to identify UPDATE and DELETE operations. If you do not enter any expressions or if no match is found for an expression, any row whose context is higher (if the sorting order is **Ascending**) or lower (if the sorting order is **Descending**) than the previous context value will be considered an INSERT.

   Note Expressions must be written in the native syntax of the ODBC source. All examples in this section are written using PostgresSQL syntax.

   - **Update expression** - Enter an expression for identifying UPDATE operations.

     Example (based on the figure Example of a Table with a Context Column):

     ```
     case when oper='U' then 1 else 0 end
     ```

     Tip: Selecting the **UPDATE the existing target record** option in the Apply Conflicts tab, eliminates the need to provide an UPDATE expression.

   - **Delete expression** - Enter an expression for identifying UPDATE operations.

     Example (based on Example of a Table with a Context Column):

     ```
     case when oper='D' then 1 else 0 end
     ```

     Important: In addition to the DELETE expression, DELETE operations should be carried out as "Soft" deletes. This means that the row is not actually deleted from the table, but rather, marked as "deleted"

Internal Parameters

Internal parameters are parameters that are not exposed in the UI and should only be used if instructed by Attunity Support.

To add internal Attunity Replicate parameters:

1. Click the **Internal Parameters** link.

   The **Internal Parameters** dialog box opens.

2. In the edit box, type the name of the parameter you need to add and then click it.
3. The parameter is added to the table below the search box with its default value.
4. Change the default value as required.
5. To reset the parameter value to its default, click the "Restore default value" icon at the end of the row.

Settings Summary

You can view a summary of your setting by clicking the **View Setting Summary** link. This is useful if you need to send a summary of your setting to Attunity Support.

Using IBM Informix as a Source

This section describes how to set up and use an IBM Informix database as the source database in a replication task.

In this section:

Prerequisites

Before you begin to work with an IBM Informix database as a source in Attunity Replicate, make sure that the following prerequisites have been met:

- **Attunity Replicate machine:**
  - Attunity Replicate installed in your network.
  - IBM Informix ODBC Driver (64-bit) version 3.70 or above must be installed on the computer where Attunity Replicate is located.
  - The `DB_LOCALE=<Informix_db_locale_value>` environment variable must be set to be the name of the IBM Informix database locale from which you want to capture data.
  - The `INFORMIXSQLHOSTS` environment variable must be set to include the names of the IBM Informix servers that are available for use as Replicate source endpoints.
- **IBM Informix Server:**
  - An IBM Informix account with the required Required Permissions.
  - CDC enabled. To enable CDC, run the script `$INFORMIXDIR/etc/syscdcv1.sql` on the IBM Informix server.

    Note This requires DBA privileges (User 'IBM Informix' or another DBA user).

  - Make sure that the database to be replicated was created with either the `WITH LOG` or the `WITH BUFFERED LOG` property.

Limitations

When using IBM Informix as a database in a Replicate task, the following limitations currently apply:

- CDC does not capture DDL changes. Due to an IBM Informix limitation, IBM Informix does not allow DDLs to be executed on tables with Full Row Logging enabled.

  To learn how to capture DDL changes during CDC, see Automatically enable full row logging.
- Due to an IBM Informix limitation, columns that follow columns of data types `SET`,`MULTISET` or `LIST` will not be replicated during CDC.

  For example, in the table below, changes to Col3 will not be captured during CDC.

Table 8.19 | Example

Name	Data Type
Col1	INTEGER
Col2	SET
Col3	INTEGER

- User-defined data types are not supported.
- Resume task by timestamp is not currently supported.

  Important: Choosing this option will resume the task from the current time.
- If a task with an IBM Informix source is stopped before any changes have been made and then resumed, any changes that were made between the time that the task was stopped and the time that it was resumed will be lost.
- Due to a known issue with the IBM Informix CDC API, Replicate does not support replication of tables whose names contain spaces or non-English letters.
- Due to a known issue with the IBM Informix transaction consistency mechanism, cached changes during Full Load are not supported.

Required Permissions

In order to access the specified database, the user specified in the **General** tab must be a member of the "IBM Informix" group (which has DBA privileges) on the database server.

IBM Informix Database Source Data Types

The following table shows the IBM Informix database source data types that are supported when using Attunity Replicate and the default mapping from Attunity Replicate data types.

For information on how to view the data type that is mapped in the target, see the section for the target database you are using.

For additional information about Attunity Replicate data types, see Replicate Data Types.

Table 8.20 | IBM Informix database Source Data Types with Mapping to Attunity Replicate Data Types

IBM Informix Source Data Types	Attunity Replicate Data Types
INTEGER	INT4
SMALLINT	INT2
INT8	INT8
SERIAL	INT4
SERIAL8	INT8
NUMERIC (p,s)	NUMERIC (p,s)
DECIMAL (p,s)	NUMERIC (p,s)
MONEY (p,s)	NUMERIC (p,s)
FLOAT	REAL8
DOUBLE	REAL8
REAL	REAL4
SMALLFLOAT	REAL4
BIGINT	STRING (20)
DATE	DATE
DATETIME (fraction)	DATETIME (fraction)
INTERVAL	STRING
CHAR	STRING (n)
VARCHAR (n)	STRING (n)
LVARCHAR (n)	STRING (n)
NCHAR (n)	STRING (n)
NVARCHAR (n)	STRING (n)
BLOB	BLOB
BYTE	BLOB
CLOB	CLOB
LIST See also Limitations.	CLOB
MULTISET See also Limitations.	CLOB

Table 8.20 | IBM Informix database Source Data Types with Mapping to Attunity Replicate Data Types (Cont.)

IBM Informix Source Data Types	Attunity Replicate Data Types
SET See also Limitations.	CLOB
TEXT	CLOB
BOOLEAN	BOOLEAN

Unsupported Data Types

The following IBM Informix data types are not supported:

- Any user-defined data type

Setting General Connection Properties

This section describes how to configure general connection properties. For an explanation of how to configure advanced connection properties, see **Setting Advanced Connection Properties** below.

Note You can also use IBM Informix files as a source. For more information, see Using the Attunity Replicate File Channel.

To add an IBM Informix source endpoint to Attunity Replicate:

1. In **Tasks** view, click **Manage Endpoint Connections** to open the **Manage Endpoints Connections** dialog box. Then click the **New Endpoint Connection** button.
2. In the **Name** field, type a name for your database. This can be any name that will help to identify the database being used.
3. In the **Description** field, type a description that helps to identify the IBM Informix database. This is optional.
4. Select **SOURCE** as the database **role**.
5. Select **IBM Informix** as the database **Type**.
6. In the **Server** field, enter the name of the IBM Informix server. On Windows, this must correspond to one of the hosts defined using the setnet32.exe tool. On Linux, this must correspond to a valid `dbservername` entry in the `$INFORMIXDIR/etc/sqlhosts` file on the computer running the application.

   Note Consider the following:
   - This information is case sensitive.
   - You can use the **Advanced** tab to add specific properties and create a custom connect string. In this case, you do not need to enter information in this tab. For more

information on using the **Advanced** tab, see Setting Advanced Connection Properties.

- To determine if you are connected to the database you want to use or if the connection information you entered is correct, click **Test Connection**.

  If the connection is successful a message in green is displayed. If the connection fails, an error message is displayed at the bottom of the dialog box.

  To view the log entry if the connection fails, click **View Log**. The server log is displayed with the information for the connection failure. Note that this button is not available unless the test connection fails.

7. Enter the IBM Informix authentication information (**User Name**, **Password**) for the authorized user for this IBM Informix database. If you do not know this information, see your IBM Informix database administrator (DBA).

   Note Consider the following:

   - This information is required. If you are using the **Advanced** tab to create a custom string, make sure to include the **User Name** and **Password** properties. See Setting Advanced Connection Properties for more information.
   - This information is case sensitive.
   - If you want to set custom properties for this database, see Setting Advanced Connection Properties.

   Important: Make sure that the IBM Informix user entered in the IBM Informix Authentication section has the correct access privileges. For information on how to provide the required privileges, see Required Permissions.

8. In the **Database name** field, enter the IBM Informix database name.

Setting Advanced Connection Properties

In the **Advanced** tab, you can set the following parameters:

- **Automatically enable full row logging**: Full Row Logging is required for CDC. Select this option to automatically enable Full Row Logging for the tables to be replicated. To automatically enable Full Row Logging, the user specified in the **General** tab must have administrative privileges on the IBM Informix database.

  Note DDL events are not captured during CDC. To perform DDL operations on source tables in a Replicate CDC task:

  - Stop the Replicate task.
  - Disable Full Row Logging for the relevant tables as in the following example:

```
execute function syscdcv1:IBM Informix.cdc_set_fullrowlogging
('sysuser:IBM Informix.employees_table', 0)
```

- Perform the DDL operation(s).
- If the **Automatically enable full row logging** option is not selected, manually enable Full Row Logging for the relevant tables.
- Start the Replicate task.
- Reload the relevant tables or perform a Full Load.

- **Max bytes per read**: Specify the maximum number of bytes to read each time the log is accessed. If you encounter performance issues, adjusting this number may help.

Internal Parameters

Internal parameters are parameters that are not exposed in the UI. You should only use them if instructed by Attunity Support.

To add internal Attunity Replicate parameters:

1. Click the **Internal Parameters** link.
   The **Internal Parameters** dialog box opens.
2. In the edit box, type the name of the parameter you need to add and then click it.
3. The parameter is added to the table below the search box with its default value.
4. Change the default value as required.
5. To reset the parameter value to its default, click the "Restore default value" icon at the end of the row.

Settings Summary

You can view a summary of your settings by clicking the **Setting Summary** link. This is useful if you need to send a summary of your settings to Attunity Support.

Using IBM DB2 for LUW as a Source

This section describes how to set up and use an IBM DB2 for LUW database as the source database in a replication task.

In this section:

Prerequisites

Before you begin to work with an IBM DB2 for LUW database as a source in Attunity Replicate, make sure that the prerequisites described in this section have been met.

Attunity Replicate Machine

The IBM DB2 for LUW client prerequisites are determined by the platform on which Attunity Replicate Server is installed (Windows or Linux).

On Windows

The IBM Data Server Client version 10.5 must be installed on the Attunity Replicate Server machine.

On Linux

The following steps need to be performed on the Attunity Replicate Server machine:

1. Download the full IBM DB2 Version 10.5 for Linux, UNIX, and Windows (**v10.5_linuxx64_server_t.tar.gz**).
2. During the installation, choose to install the IBM DB2 Data Server Driver for ODBC and CLI *only*.
3. Add the DB2 driver location to the Linux library path (if not already there), by running the following Shell command:

   Syntax:

   ```
   export LD_LIBRARY_PATH=$LD_LIBRARY_PATH:path/lib64
   ```

   Where *path* is the path to the driver.

   Example:

   ```
   export LD_LIBRARY_PATH=$LD_LIBRARY_PATH:/opt/ibm/db2/V10.5/lib64
   ```
4. Edit the file **/etc/odbcinst.ini** and add this section:

   ```
   [IBM DB2 ODBC DRIVER]
   ```

```
Driver = /opt/ibm/db2/V10.5/lib64/libdb2o.so
fileusage=1
dontdlclose=1
```

5. Restart Attunity Replicate Server.

IBM DB2 for LUW Server

To enable CDC (Change Data Capture):

- **Set the database to be recoverable** - To capture changes, Replicate requires that the database is configured to be recoverable. a database is recoverable if either or both of the database configuration parameters `LOGARCHMETH1` and `LOGARCHMETH2` are *not* set to OFF.
- **Permissions** - The Attunity user must be granted the following permissions:
  - SYSADM or DBADM
  - DATAACCESS

Limitations

When using IBM DB2 for LUW as a database in a Replicate task, the following limitations currently apply:

- Clustered database is not supported

  Note Users can define a separate IBM DB2 for LUW database for each of the endpoints in the cluster.

- **Change processing limitations:**
  - When truncating a table with multiple partitions, the number of DDL events displayed in the Replicate console will be equal to the number of partitions. This is because IBM DB2 for LUW records a separate DDL for each partition.
  - The following DDLs on partitioned tables are not supported:
    - `ALTER TABLE ADD PARTITION`
    - `ALTER TABLE DETACH PARTITION`
    - `ALTER TABLE ATTACH PARTITION`
  - Change processing does not support the DECFLOAT data type. Consequently, changes to DECFLOAT columns will be ignored during CDC.
  - The RENAME COLUMN statement is not captured.
  - When performing updates on MDC (Multi-Dimensional Clustering) tables, each update is shown in the Replicate console as INSERT + DELETE.
  - When the task setting "Include LOB columns in replication" is disabled, any table that has LOB columns will be suspended during change processing.
  - When the **Audit table** option is enabled in the **Store Changes Settings** tab, the first timestamp record in the audit table will be NULL.

- When the **Change table** option is enabled in the **Store Changes Settings** tab, the first timestamp record in the table will be Zero (i.e. 1970-01-01 00:00:00.000000).
- **DB2 10.5 and above:** Variable-length string columns with data that is stored out-of-row will be ignored. Note that this limitation is only applicable to tables created with extended row size.

IBM DB2 for LUW Database Source Data Types

The following table shows the IBM DB2 for LUW database source data types that are supported when using Attunity Replicate and the default mapping from Attunity Replicate data types.

For information on how to view the data type that is mapped in the target, see the section for the target database you are using.

For additional information about Attunity Replicate data types, see Replicate Data Types.

Table 8.21 | IBM DB2 for LUW database Source Data Types with Mapping to Attunity Replicate Data Types

IBM DB2 for LUW Source Data Types	Attunity Replicate Data Types
INTEGER	INT4
SMALLINT	INT2
BIGINT	INT8
DECIMAL (p,s)	NUMERIC (p,s)
FLOAT	REAL8
DOUBLE	REAL8
REAL	REAL4
DECFLOAT (p)	If precision = 16, then: REAL8 If precision is = 34, then: STRING
GRAPHIC	WSTRING n<=127
VARGRAPHIC	WSTRING n<=16k double byte chars
LONG VARGRAPHIC	CLOB
CHAR (n)	STRING n<=255
VARCHAR (n)	STRING

Table 8.21 | IBM DB2 for LUW database Source Data Types with Mapping to Attunity Replicate Data Types (Cont.)

IBM DB2 for LUW Source Data Types	Attunity Replicate Data Types
	n<=32k
LONG VARCHAR (n)	CLOB
	n<=32k
CHAR (n) FOR BIT DATA	BYTES
VARCHAR (n) FOR BIT DATA	BYTES
LONG VARCHAR FOR BIT DATA	BYTES
DATE	DATE
TIME	TIME
TIMESTAMP	DATETIME
BLOB	BLOB
CLOB	CLOB
	Maximum size: 2 GB
DBCLOB	CLOB
	Maximum size: 1 G double byte chars
XML	CLOB

Setting General Connection Properties

This section describes how to configure general connection properties. For an explanation of how to configure advanced connection properties, see **Setting Advanced Connection Properties** below.

Note You can also use IBM Informix files as a source. For more information, see Using the Attunity Replicate File Channel.

To add an IBM Informix source endpoint to Attunity Replicate:

1. In **Tasks** view, click **Manage Endpoint Connections** to open the **Manage Endpoints Connections** dialog box. Then click the **New Endpoint Connection** button.
2. In the **Name** field, type a name for your database. This can be any name that will help to identify the database being used.
3. In the **Description** field, type a description that helps to identify the IBM Informix database. This is optional.
4. Select **SOURCE** as the database **role**.
5. Select **IBM Informix** as the database **Type**.
6. In the **Server** field, enter the name of the IBM Informix server. On Windows, this must

correspond to one of the hosts defined using the setnet32.exe tool. On Linux, this must correspond to a valid `dbservername` entry in the `$INFORMIXDIR/etc/sqlhosts` file on the computer running the application.

Note Consider the following:

- This information is case sensitive.
- You can use the **Advanced** tab to add specific properties and create a custom connect string. In this case, you do not need to enter information in this tab. For more information on using the **Advanced** tab, see Setting Advanced Connection Properties.
- To determine if you are connected to the database you want to use or if the connection information you entered is correct, click **Test Connection**.

  If the connection is successful a message in green is displayed. If the connection fails, an error message is displayed at the bottom of the dialog box.

  To view the log entry if the connection fails, click **View Log**. The server log is displayed with the information for the connection failure. Note that this button is not available unless the test connection fails.

7. Enter the IBM Informix authentication information (**User Name**, **Password**) for the authorized user for this IBM Informix database. If you do not know this information, see your IBM Informix database administrator (DBA).

Note Consider the following:

- This information is required. If you are using the **Advanced** tab to create a custom string, make sure to include the **User Name** and **Password** properties. See Setting Advanced Connection Properties for more information.
- This information is case sensitive.
- If you want to set custom properties for this database, see Setting Advanced Connection Properties.

Important: Make sure that the IBM Informix user entered in the IBM Informix Authentication section has the correct access privileges. For information on how to provide the required privileges, see Required Permissions.

8. In the **Database name** field, enter the IBM Informix database name.

Setting Advanced Connection Properties

In the **Advanced** tab, you can set the following parameters:

- **Automatically enable full row logging**: Full Row Logging is required for CDC. Select this option to automatically enable Full Row Logging for the tables to be replicated. To automatically enable Full Row Logging, the user specified in the **General** tab must have

administrative privileges on the IBM Informix database.

Note DDL events are not captured during CDC. To perform DDL operations on source tables in a Replicate CDC task:

- Stop the Replicate task.
- Disable Full Row Logging for the relevant tables as in the following example:

```
execute function syscdcv1:IBM Informix.cdc_set_fullrowlogging
('sysuser:IBM Informix.employees_table', 0)
```

- Perform the DDL operation(s).
- If the **Automatically enable full row logging** option is not selected, manually enable Full Row Logging for the relevant tables.
- Start the Replicate task.
- Reload the relevant tables or perform a Full Load.

- **Max bytes per read**: Specify the maximum number of bytes to read each time the log is accessed. If you encounter performance issues, adjusting this number may help.

Internal Parameters

Internal parameters are parameters that are not exposed in the UI. You should only use them if instructed by Attunity Support.

To add internal Attunity Replicate parameters:

1. Click the **Internal Parameters** link.
   The **Internal Parameters** dialog box opens.
2. In the edit box, type the name of the parameter you need to add and then click it.
3. The parameter is added to the table below the search box with its default value.
4. Change the default value as required.
5. To reset the parameter value to its default, click the "Restore default value" icon at the end of the row.

Settings Summary

You can view a summary of your settings by clicking the **Setting Summary** link. This is useful if you need to send a summary of your settings to Attunity Support.

Using IBM DB2 for iSeries as a Source

This section describes how to set up and use an IBM DB2 for iSeries database as the source database in a replication task.

In this section:

Prerequisites

The following topic lists the prerequisites for using an IBM DB2 for iSeries endpoint in a Replicate task.

Client

- **Windows:** To work with an IBM DB2 for iSeries database as a source in Attunity Replicate, the iSeries ODBC driver (5770-XE1 IBM i Access for Windows) must be installed on the Replicate Server machine.

  The minimum supported version is 12.64.

  The driver is part of IBM i Access Client Solutions, which can be downloaded from the IBM web site by authorized IBM customers. Note that it is not necessary to install all of the IBM i Access Client Solutions components, only the ODBC driver (which is installed regardless of which components you choose to install).

- **Linux:** To work with an IBM DB2 for iSeries database as a source in Attunity Replicate, the iSeries ODBC driver (5770-XL1 IBM i Access for Linux) must be installed on the Replicate Server machine.

  The minimum supported version is 12.64.

  The driver is part of IBM i Access Client Solutions, which can be downloaded from the IBM web site by authorized IBM customers. Note that it is not necessary to install all of the IBM i Access Client Solutions components, only the ODBC driver (which is installed regardless of which components you choose to install).

  After installing the Linux Driver, edit the file /etc/**odbcinst.ini** and add the following section:

  ```
  [IBM i Access ODBC Driver 64-bit]
  Description = IBM i Access for Linux 64-bit
  ODBC Driver Driver = /opt/ibm/iaccess/lib64/libcwbodbc.so
  ```

```
Setup = /opt/ibm/iaccess/lib64/libcwbodbcs.so
Threading = 0
DontDLClose = 1
UsageCount = 1
```

Change Processing

- For DB2 iSeries version 7.1, the following IBM patch needs to be installed:

  DB2 for IBM i Group PTF

  SF99701: 710 DB2 for IBM i - Level 3 (released August 2010) or PTF '5770SS1 V7R1M0 SI39820' , PTF '5770SS1 V7R1M0 SI39821'.
- All of the source tables for a given Replicate task need to be journaled to *the same journal*. The name of the journal and the library in which it is located must be specified in the endpoint settings. During the task, Replicate polls this journal for changes to the source tables.

  When you start journaling the source tables, the `Record images` parameter can be set to `*BOTH` (for capturing before and after images) or `*AFTER`.

  Note If you need to run several Replicate tasks (that replicate data from IBM DB2 for iSeries), it is more efficient (though not essential) to create a separate journal for each task. As only one journal can be specified per endpoint, you would also need to define a separate endpoint for each task.

Required Permissions

The following permissions must be granted to the user specified in the **General** tab of the IBM DB2 for iSeries endpoint settings:

- USER CLASS = *USER (default value)
- Special authority = *NONE
- Full Load: Read permissions for the source tables.
- CDC: Read permissions for the journal defined for the IBM DB2 for iSeries endpoint and for the task's source tables.

You must also set the following **Authorities and Locks** for the IBM DB2 for iSeries database:

- **Journal Authority:** *USE
- **Journal Library Authority:** *EXECUTE
- **Journal Receivers Authority:** *USE
- **Journal Receivers Library's Authority:** *EXECUTE

- **File Authority (if specified):** *USE
- **File Library Authority:** *EXECUTE
- **Journal Lock:** *SHRRD
- **Journal Receiver Lock:** *SHRRD
- **File Lock (if specified):** *SHRRD

*OBJEXIST is also required for the journal authority if any of the following are true:

- *ALLFILE has been specified for the file key.
- Specified object does not exist on the system.
- *IGNFILSLT or *IGNOBJSLT is specified for the journal code selection value for any selected journal codes.
- The journal is a remote journal.

Limitations

The following limitations apply when using the IBM for DB2 iSeries endpoint as a source in a Replicate task:

- Field level encryption is not supported
- Journal receivers must be System Managed rather than User Managed.
- All tables must be in the same journal
- The DROP TABLE DDL is not supported
- Limited LOB support is actually full LOB support
- Replicating BLOBs will significantly affect performance
- The XML data type is not supported
- The DBCLOB data type is not supported
- Row size cannot exceed 32740 bytes
- Partitioned tables are not supported
- When the **Use table and schema system names** option is enabled (in the **Advanced** tab), only table/schema names that contain uppercase letters, digits, and underscores (_) are supported.

  For more information on this option, see Replicating System Names.

IBM DB2 for iSeries Database Source Data Types

The following table shows the IBM DB2 for iSeries database source data types that are supported when using Attunity Replicate and the default mapping from Attunity Replicate data types.

For information on how to view the data type that is mapped in the target, see the section for the target database you are using.

For additional information about Attunity Replicate data types, see Replicate Data Types.

Table 8.22 | IBM DB2 for iSeries database Source Data Types with Mapping to Attunity Replicate Data Types

IBM DB2 for iSeries Source Data Types	Attunity Replicate Data Types
INTEGER	INT4
SMALLINT	INT2
BIGINT	INT8
DECIMAL (p,s)	NUMERIC (p,s)
FLOAT	REAL8
DOUBLE	REAL8
REAL	REAL4
CHAR (n)	If n<=32 KB, then: STRING
VARCHAR (n)	If n<=32 KB, then: STRING
GRAPHIC (n)	If n<=16 KB, then: STRING
VARGRAPHIC (n)	If n<=16 KB double byte chars, then: STRING
DATE	DATE
TIME	TIME
TIMESTAMP	DATETIME (6)
BLOB	BLOB Maximum size: 2 GB
CLOB	CLOB Maximum size: 2 GB
DBCLOB	CLOB Maximum size: 1 GB double byte chars
ROWID	BYTES - This should be a user-defined column.
DATALINK	STRING
TIMESTAMP WITH TIME ZONE	NOT SUPPORTED

Setting General Connection Properties

This section describes how to configure general connection properties. For an explanation of how to configure advanced connection properties, see **Setting Advanced Connection Properties** below.

Note You can also use IBM DB2 for iSeries File Channel files as a source. For more information, see Using the Attunity Replicate File Channel.

To add an IBM DB2 for iSeries source endpoint to Attunity Replicate:

1. In the Attunity Replicate console, click **Add database** to open the Add Endpoints dialog box.
2. In the **Name** field, type a name for your database. This can be any name that will help to identify the database being used.
3. In the **Description** field, type a description that helps to identify the IBM DB2 for iSeries database. This is optional.
4. Select **SOURCE** as the database **role**.
5. Select **IBM DB2 for iSeries** as the database **Type**.
6. Choose one of the following:
   - **Use ODBC DSN** - If you choose this option, specify the IBM DB2 for iSeries database ODBC DSN.
   - **Use these connection properties** - If you choose this option, enter the IBM DB2 for iSeries **Server** (hostname or IP address).
7. Enter the IBM DB2 for iSeries authentication information (**User Name**, **Password**) for the authorized user for this IBM DB2 for iSeries database. If you do not know this information, see your IBM DB2 for iSeries database administrator (DBA).

   Note This information is case sensitive.

   Important: Make sure that the specified user has the required access privileges.

8. In the **Journal Name** field, enter the name of the journal containing the source tables.

   See also: Change Processing prerequisites.
9. In the **Journal Library** field, enter the name of the library where the journal is located.

   See also: Change Processing prerequisites.

Setting Advanced Connection Properties

In the **Advanced** tab, you can set advanced connection properties such as overriding CCSID to Character Set mapping and setting internal Replicate parameters.

Overriding CCSID to Character Set Mapping

In some cases, character data in source tables may be encoded in a different CCSID than what is declared in the source database table definition. For example, a specific table or column definition might indicate that its uses CCSID 500 (EBCDIC International) whereas in fact, it uses CCSID 1148 (ENCDIC International with EURO). In this case, you can tell Replicate that the source definition CCSID 500 should be treated as CCSID 1148 (specifically, the character set named IBM-1148).

Note that when the source table definition specifies CCISD 65535 (meaning character set is unknown), you must specify what character set should be assumed when reading data from that table or column.

Note If there is a conflict between the character set mapping for a specific column and the character set mapping defined in the endpoint settings, the column-level character set mapping takes precedence.

For more information on overriding character set mapping at column level, see Using the Transform Tab.

To do this:

1. In the **Override CCSID to Character Set Mapping** section, click the **New** button.
   A row is added to the table.
2. Enter the CCSID in the **CCSID** column and the code page in the **Character set** column.
3. Repeat to map additional CCSID values.

Converting to a Custom Code Page

Perform the following procedure if your source endpoint tables are defined with an incorrect CCSID and the correct definition is actually in a UCM file.

1. Create or download a mapping data file with the file extension **.ucm**.
2. Create a CNV file for the UCM file by running the following command:
   `<product_dir>\bin\makeconv.exe -v <file_name>.ucm`
   Example:
   `"c:\Program Files\Attunity\Replicate\bin\makeconv.exe" -v 1047_EX.ucm`
   This will create a CNV file with the same name as the UCM file (e.g. **1047_EX.cnv**).
3. Copy the file to the following location:
   `<product_dir>\bin\icudt58l`
4. Add a new character set mapping as follows:
   a. In **CCSID** column, enter the original source CCSID number (e.g. **1047**)
   b. In the **Character set column**, enter the name of the CNV file without the extension (e.g. **1047_EX**).

Adding the RRN Column to Target Tables

Source tables that do not have a primary key, a unique index, or a combination of columns that can be used as a unique index, must be registered using the relative record numbers (RRN).

Select one the following options:

- Do not add RNN column to target tables
- Add RRN column to target tables without a primary key or unique index
- Add RRN column to all target tables

When you select one of the "Add RRN columns" options, both the Change Tables and the target tables will have an extra column, ATTREP_RRN of type INTEGER, which contains a unique value for each row. This column contains the RRN that corresponds to each source table row.

Replicating System Names

The IBM DB2 for iSeries source endpoint replicates tables based on their SQL names (unlimited length). If your IBM DB2 for iSeries database does not use SQL names, it's likely that you'll want to keep the replication based on system names.

To do this, select the **Use table and schema system names** check box.

Skipping Journal Validation

From IBM DB2 for iSeries 7.3, Replicate automatically validates the specified journal. This involves checking that the journal exists and that it contains the tables selected for replication. When numerous tables are selected for replication, this process may take some time. In such cases, if you are sure that the specified journal exists and that it contains the correct tables, you may want to skip the validation phase.

To do this, select the **Skip journal validation** check box.

Internal Parameters

Internal parameters are parameters that are not exposed in the UI. You should only use them if instructed by Attunity Support.

To add internal Attunity Replicate parameters:

1. Click the **Internal Parameters** link.

   The **Internal Parameters** dialog box opens.
2. In the edit box, type the name of the parameter you need to add and then click it.
3. The parameter is added to the table below the search box with its default value.
4. Change the default value as required.

5. To reset the parameter value to its default, click the "Restore default value" icon at the end of the row.

Settings Summary

You can view a summary of your settings by clicking the **Setting Summary** link. This is useful if you need to send a summary of your settings to Attunity Support.

Using IBM DB2 for z/OS as a Source

This section describes how to set up and use an IBM DB2 for z/OS database as the source endpoint in a replication task.

In this section:

Prerequisites

The following topic lists the prerequisites for working with the Attunity Replicate IBM DB2 for z/OS endpoint.

In this topic:

ODBC Requirements

The Attunity Replicate IBM DB2 for z/OS source endpoint relies on the *IBM Data Server Driver for ODBC* for access to data, Change Data and metadata. This section describes the client side and server side ODBC prerequisites.

Client Side ODBC Requirements (Linux, Unix and Windows)

Install "IBM Data Server Driver for ODBC and CLI" for DB2 11.1 for LUW on the Attunity Replicate Server machine.

Note Using DB2 10.5.400 for LUW may be possible with a few modifications to the DB2CLI.INI file. If you want to use this version, please consult with Attunity support.

Server Side ODBC Setup

Bind the plan to be used for ODBC, as specified in the `PLANNAME=` value in the ODBC initialization file. The default name is `DSNACLI`. The BIND job can be found in member `DSNTIJCL`, which is in the `SDSNSAMP` library of the source DB2 installation.

Use the DB2CLI bind command to bind the ODBC-supplied packages to your intended source z/OS DB2 subsystem. This action is described in Configuring your developer and

runtime environment on the IBM website. For information about the DB2CLI utility, including an example for the bind utility, see db2cli - DB2 interactive CLI command on the IBM website.

IBM Data Server Driver for JDBC and SQLJ

Follow the instructions in the DB2 online help.

Required Permissions

To enable Replicate to extract data from the source tables (Full Load and Change Data Capture), the user specified in the IBM DB2 for z/OS endpoint settings must be granted the following permissions:

- EXECUTE on the IFI reading the stored procedure (only required for Change Data Capture)
- SELECT on the source tables and on the DB2 catalog tables
- MONITOR2 to be able to start IFI sessions (only required for Change Data Capture)

For additional details about these permissions, see Change Data Capture Requirements.

Change Data Capture Requirements

To capture changes from IBM DB2 for z/OS, Attunity Replicate uses a special program - invoked as an external routine - which can either be a stored procedure or a user-defined table function. This program (a load module) as well as the stored procedure definition need to be installed and configured on the z/OS system before changes can be captured. The installation procedure, which should be performed by the DBA, is described below.

Additionally, the Data Capture Changes attribute must be set for every table whose changes you want to replicate. You can either do this manually or allow Replicate to do this by leaving the **Automatically enable Data Capture Changes (requires admin privileges)** option enabled (the default) in the **Advanced** tab.

Receiving the Load Modules and JCLs

The following section explains how to download and receive the Attunity Replicate IBM DB2 for z/OS endpoint installation kit.

To receive the load modules and JCLs

1. Download the **AttunityReplicate_<version>_r4db2.zip** file from the Customer Zone or obtain it from your Attunity Sales representative. Once downloaded, unzip the file contents to your workstation.

   The zip file contains the following files:

   - `install.xmit`
   - `load.xmit`
   - `readme.txt`
   - `switch.txt`

2. Allocate the `<receiveHLQ>.LOAD.XMIT` and `<receiveHLQ>.INSTALL.XMIT` datasets in z/OS with the following attributes:

   `LRECL=80, RECFM=FB, DSORG=PS.`

3. Transfer `LOAD.XMIT` from your workstation to `<receiveHLQ>.INSTALL.XMIT` in z/OS using binary transfer, and then transfer `install.xmit` to `<receiveHLQ>.INSTALL.XMIT`.

4. From TSO, customize the following job, and then submit it:

```
//*******************************************************************
//*                     Sample JCL for receiving                    *
//*  Attunity ReplicateIBM DB2 for z/OS endpoint installation kit   *
//* 1. Add a jobcard                                                *
//* 2. Replace all <receiveHLQ> by the High Level Qualifier used    *
//*    for receive files                                            *
//* 3. Replace all <r4z-hlq> by the High Level Qualifier chosen for *
//*    the installation files                                       *
//*******************************************************************
//RCVLOAD  EXEC PGM=IKJEFT01,DYNAMNBR=20
//SYSPRINT DD SYSOUT=*
//SYSTSPRT DD SYSOUT=*
//SYSTSIN DD *
RECEIVE INDSN('<receiveHLQ>.LOAD.XMIT') NODISPLAY
DATASET('<r4z-hlq>.LOAD')
/*
//RCVINSTL EXEC PGM=IKJEFT01,DYNAMNBR=20
//SYSPRINT DD SYSOUT=*
//SYSTSPRT DD SYSOUT=*
//SYSTSIN DD *
RECEIVE INDSN('<receiveHLQ>.INSTALL.XMIT') NODISPLAY
DATASET('<r4z-hlq>.INSTALL')
/*
```

5. After customizing and running the job, make a note of the Attunity Replicate IBM DB2 for z/OS endpoint installation kit dataset names.

Installing the R4DB2 z/OS Component

Note The configuration steps below include submitting jobs found in the INSTALL library (referred to as **`<r4z-hlq>.INSTALL`** in the section above). Each job requires the INSTALL library (or its copy) as its JCLLIB, and must have both a job card and JES control statements that comply with the site's regulations.

Step 1: Global Changes in the INSTALL Library

During the install process, you will be required to submit most of the members prefixed "DO*" in the library. Therefore, before submitting any job, it is recommended to update all "DO*" jobs with the following changes:

- Set the job name, the job card parameters and JES control statements to comply with your system requirements.

- Set the library specified in the JCLLIB ORDER parameter as the INSTALL library.

Step 2: Defining and Enabling the R4Z Executables

Edit DFSYMLIST which contains the JCL symbols to be used by jobs in this library. These symbols are listed in Table 8.23 Installation Symbols.

The following symbols should be set according to values chosen in Step 1: Global Changes in the INSTALL Library:

- **&R4ZHLVL** must be assigned as the high-level-qualifier of the R4Z INSTALL and LOAD libraries, i.e. the longest common qualifier of those libraries, referred to as <r4z-hlq> in Receiving the Load Modules and JCLs.
- **&R4ZLOADL** must be assigned with the R4Z LOAD library name, or left unchanged if it is identical to &R4ZHLVL..LOAD.
- **&R4ZINSTL** must be assigned with the R4Z INSTALL library name, or left unchanged if it is identical to &R4ZHLVL..INSTALL
- **&R4ZUSER** must be a TSO user-ID which is granted access to DB2. This user-ID will be used to establish an ODBC connection with DB2 from the client environment.

The rest of the symbols must be set according to products already installed in the system, or (if prefixed &R4Z*) according to values desired by the installer.

Choose appropriate values for all symbols. Note that some variables are dependent on previous ones. Make sure not to change the order of instructions.

1. Set the LOAD library as APF-authorized:

   Submit jobs/complete actions in members prefixed DO1APF* to set the library specified in &R4ZLOADL as APF-authorized.

   Jobs should terminate with RC 0.

   - **Prerequisites:** The APF list format must be DYNAMIC for job is run.
   - **Authorizations:**
     - A resource profile named CSVAPF.<r4z-hlq>.LOAD must be RDEFINEd in class FACILITY, or a generic profile covering that name must exist.
     - The job submitter must get PERMIT on that profile. Note that it is possible to give APF control over any library, using profile CSVAPF.**.

2. Set a WLM application environment where the stored procedure/UDTF will process:

   Submit jobs/complete actions in members prefixed DO1WLM* to define the WLM application environment, define a JCL procedure for its started task, and provide RACF privileges for the user-ID assigned as the OWNER for the started tasks in executes.

   Jobs should terminate with RC 0 or 4.

   - **Prerequisites:**
     - It is possible to modify the WLM policy. To verify this, ensure that job DSNTWLMB in the DB2 SDSNSAMP library (which is used to define WLM application environments) can run.
     - You need to set the appropriate value for symbol &STCUSER, which is the default user-ID assigned to STCs (started tasks) by the JCL procedure name. It may be

identical to the OWNER of the other STCs in the DB2 subsystem. If so, see that owner in the SDSF "display active". If you are unsure of the default user-ID, use a default STCUSER in the job and search the system log for a ICH408I message with `CL(SERVER)`. The user in that message will be specified as `USER(…)`.

For more information about STC user-IDs, refer to the IBM online documentation.

- **Authorizations:**
  - READ and UPDATE on `MVSADMIN.WLM.POLICY`, to query and update the WLM policy.
  - UPDATE on the procedure library specified in `DSSYMLST` in the `&PROCLIB` symbol.
  - Perform RACF commands (`RDEFINE`, `PERMIT`) on class SERVER. If you are not sure how to do this, consult with your system programmer.

3. Create a stored procedure/table-function in DB2, and grant execution on it:

   Submit jobs/complete actions in members prefixed `DO3SP*` to CREATE the stored procedure/table-function that performs change-capture. If the procedure already exists, DROP will precede the CREATE; if the user which is to be used for ODBC access (`&R4ZUSER` symbol) is not the user submitting the job, it will also GRANT execution on the procedure to the specified user.

   Jobs should terminate with `RC 0` or `4`.

   - **Authorizations:**

     The submitting user must either have one of the following authorizations:
     - SYSADM or SYSCTRL authority.
     - CREATEIN authorization in the schema designated by `&R4ZSCNM`.

4. Grant the required monitoring and metadata privileges to ODBC user-ID:

   Assuming the user specified for `&R4ZUSER` (the ODBC user) is *not* your user-ID, submit job in member `DO4GRANT` to grant the ODBC program sufficient authorization to access DB2 metadata and to start the CDC trace required for the data capture process.

   This will:
   - GRANT SELECT to the ODBC user on the following tables:
     - SYSIBM.SYSTABLES
     - SYSIBM.SYSTABLESPACE
     - SYSIBM.SYSCOLUMNS
     - SYSIBM.SYSROUTINES
     - SYSIBM.SYSDUMMY1
   - ALTER SYSIBM.SYSTABLES enabling DATA CAPTURE.

   Jobs should terminate with `RC 0`.

   - **Authorizations:**

     The submitting user must either have one of the following authorizations:
     - SYSADM or SYSCTRL authority.
     - TRACE/ MONITOR2 with grant option.

5. To verify that the R4Z product is configured correctly, you can test Change Processing on

user tables. This is done as follows:

Submit jobs/complete actions in members prefixed `DO5AC*` to grant the ODBC program sufficient authorization to access DB2 user tables, and set data capture to ON. This job may be run several times, each time per another set of user tables. To save the performed actions (for a possible UNDO requirement), set a unique `DSEQ` parameter for each run.

- **Authorizations:**

  The submitting user must have one of the following authorizations:

  - SYSADM or SYSCTRL authority
  - DBADM on the database(s) in which the designated tables are defined
  - SELECT WITH GRANT OPTION authorization (to GRANT SELECT) in addition to the ALTER authorization (to set the tables with DATA CAPTURE CHANGES) on the designated tables

Step 3: Establishing R4Z Logical Environments.

Note Before performing this step, it is recommended that you read the concepts of the R4Z CDC processing, as described in Controlling the CDC Process.

Deciding on the logical environments: Based on predicting the need for independent replication processes (to allow different versions, to ensure sufficient ECSA memory, to account for different latencies allowed, etc.) – Determine which R4Z CDC environments are to exist in each LPAR. For each environment, choose the R4Z environment suffix names (up to 8 letters) and the shorthand (a 3-letter identifier). The shorthand is merely for good housekeeping, as will be made clear later.

The INSTALL library contains an include-member `DFENVTS1`, which specifies an R4Z environment with suffix `TEST1`; and sample jobs XM*, using shorthand: TS1 to refer to it. Use these as an example to create R4Z environments with suffix and shorthand qualifiers of your choice. Then, for each R4Z environment you want to define, perform the following steps:

1. Select an identifier of up to 8 letters, `<env-suffix>`, as the suffix of the logical environment name.
2. Choose a 3-letter shorthand, `<esh>` for the environment suffix you have selected.
3. Create member `DFENV<esh>`. Then, in the member, code a JCL SET instruction as follows:

   ```
   // SET R4ZENV=<env-suffix>
   ```

   Choose appropriate values for the rest of the JCL symbols. The symbols are documented in Table 8.24 Control Program Symbols below.

   Use member `DFENVTS1` as a sample.
4. To enable CDC processing, create member `ITCDC<esh>` to establish R4Z environment structures in ECSA (and name/token pairs addressing them).

   Use job `ITCDCENV` as a sample.

5. Set up an automation facility to run job `ITCDC<esh>`, which reestablishes the environment, each time the system (LPAR) is brought up.

Table 8.23 | Installation Symbols

Symbol Name	Description	Attribute Default Value
&SSID	The subsystem-ID of the source DB2.	4-character name. DSN1
&DB2HLVL	High-level qualifier of DB2 installation libraries, not including SSID qualifier.	<qualifier>.<qualifier>… DSNB10
&DB2LOADL	High-level qualifier of DB2 SDSNLOAD library.	Library name. &DB2HLVL..&SSID..SDSNLOAD
&DB2EXITL	High-level qualifier of DB2 SDSNEXIT library.	Library name. &DB2HLVL..&SSID..SDSNEXIT
&DB2RUNL	High-level qualifier of DB2 RUNLIB.LOAD library.	Library name. &DB2HLVL..&SSID..RUNLIB.LOAD
&DB2VER	The z/OS DB2 version.	Number (10..12) 11
&CEERUNL	z/OS LE (language environment) and C++ runtime library.	Library name. CEE.SCEERUN
&PROCLIB	JCL procedure library where third-party JCL procedures are located. You can choose a PROCLIB used by third-party products.	Library name. N/A
&R4ZHLVL	Replicate for z/OS high-level qualifier.	<qualifier>.<qualifier>… R4DB2
&R4ZLOADL	Replicate for z/OS load library.	Number (09..12). &R4ZHLVL..LOAD
&R4ZWLMPR	WLM STC for the Replicate application environment.	Member name (JCL procedure). &SSID.WR4Z
&R4ZAPPNV	The WLM application environment serving the Replicate stored procedure.	Upper case name, up to 32 characters (underscores allowed).

Table 8.23 | Installation Symbols (Cont.)

Symbol Name	Description	Attribute / Default Value
		R4DB2WLM
&STCUSER	The user ID assigned to regions started by WLM application-environment	User ID (up to 7 characters).
		START1
&RAZUSER	User ID for calling the stored procedure (to be specified in the Replicate endpoint connection settings).	User ID (up to 7 characters).
		ATTUSER
&R4ZSCNM	Schema name qualifying the stored procedure routine (to be specified in the Replicate endpoint connection settings).	Schema name.
		&R4ZUSER
&R4ZIFISP	Stored procedure name (to be specified in the Replicate endpoint connection settings).	Routine name (not including the schema name).
		R4DB2SP
&R4ZIFITF	User-defined function name (to be specified in the Replicate endpoint connection settings).	Function name (not including the schema name).
		R4ZUDF

Table 8.24 | Control Program Symbols

Symbol Name	Description	Attribute Default Value
&R4ZENV	A logical environment for Replicate to control its tasks in isolation from other environments. The name provided is a suffix to the default environment name: R4DB2.	Identifier of up to five characters. R4DB2
&MAXSESS	MAXSESSIONS parameter of R4DB2CTL. Number of IFI concurrent sessions allowed.	Number: 1..128. 32
&MAXIFISZ	MAXIFIBUFSIZE parameter of R4DB2CTL. Maximal buffer size to be allocated in ECSA for IFI records.	Number in K-bytes: 64 to 1024 in multiples of 32. 512
&MAXIDLE	MAXIDLEMINS parameter of R4DB2CTL. Time limit for an IFI session to remain idle, before being closed.	Number of minutes: up to 120. Should be below IDTHTOIN parameter in ZPARM . 5
&MAXLOGP	MAXIFILOGPOSSECS parameter of R4DB2CTL. Time after which IFI open read sessions are terminated. An open session keeps DBD locks.	Number of minutes: up to 120. &DB2HLVL..&SSID..RUNLIB.LOAD
&MAXRESLT	MAXRESULTSET parameter of R4DB2CTL. Maximum accumulated size of rows returned by user-defined table function.	Number in megabytes: 1..32. 4

Limitations

The following limitations apply when using the IBM DB2 for z/OS endpoint in a Replicate task:

- During a task with Full Load and Apply Changes enabled, duplicate keys may be replicated to the target. This is because records that were updated and cached during the Full Load stage of the task may have the same timestamp as the original records that were inserted during Full Load. Note that this usually only happens when the time difference between the inserted record and the updated (cached) record is very small (milliseconds).

Controlling the CDC Process

Attunity Replicate uses ECSA memory structures - called R4Z environments - which are formed during CDC processing. Each z/OS LPAR may contain several such R4Z environments. To form the R4Z environments, a special utility program, R4DB2CTL, is provided.

This program can also be used to manage the state of CDC processing as well as the level of traffic on the z/OS side.

The main uses for the R4DB2CTL program are (relevant command):

- Initialize or terminate the R4DB2 environment on a z/OS LPAR (`SETACTIVE`, `SETINACTIVE`, `TERMINATE`, `FORCE`)
- Query current sessions and their operational parameters (no parameter)
- Alter operational parameters and limits of R4DB2 (`SETACTIVE`)
- Suspend or resume a session associated with a specific Replicate task (`SUSPEND`, `RESUME`)
- Dump R4DB2 internal structures for troubleshooting (`DUMP`)

As the R4DB2CTL programs requires APF-authorization, all libraries in the STEPLIB must be APF- authorized.

When running R4DB2CTL with no parameter, its completion-code is set based on whether the environment was initialized (`CC=0`) or not initialized (`CC=1`).

Invocation Syntax

The `R4DB2CTL` program is invoked as a job step, i.e. `EXEC PGM=R4DB2CTL`, and accepts instructions via the invocation parameter of the job step. For readability, most sample jobs provided in the INSTALL library are implementing a rather recent option (as of z/OS 2.1) for specifying the invocation parameters – instead of using the `PARM` parameter, as in:

```
PARM='argument1,argument2,…'
```

Example of PARMDD Parameter Usage:

```
//stepname EXEC PGM=program,PARMDD=MYPARMS
//MYPARMS DD *
argument1,
argument2,
…
/*
```

The `R4DB2CTL` invocation parameter is a string comprising of optional sub-parameters, separated by a comma as follows:

```
[ENV=environment,][MSGLVL={0 | 1},][action]
```

Note The order of the string is not important.

Where:

ENV=environment designates the logical scope of Replicate activity, upon which the control program is to act. Managing several Replicate scopes in a single LPAR may be necessary if there are insufficient policies for allocating resources to Replicate tasks, or if there are more than one release of the R4DB2 product.

For more information on the **ENV=environment** parameter, see Installing the R4DB2 z/OS Component.

MSGLVL={0 | 1 | 2} designates the level of notifications to be displayed in the message file during the operation of the control program.

- 0 = No notifications
- 1 = Moderate
- 2 = Maximum

action can be one of the following:

SETACTIVE(***session-limits***)

Enable replication CDC sessions. The ***session-limits*** optional qualifier includes thresholds controlling resources used for CDC processing, and is described in Syntax Elements below.

SETINACTIVE

Disables all replication sessions. The R4DB2 statue is retained.

PAUSE(***instance-qualifier***)

Suspends CDC retrieval for all instances matching the ***session-limits*** qualifier. Replication tasks suspended for more than a certain time are stopped will attempt recovery multiple times.

RESUME(***instance-qualifier***)

Resumes CDC retrieval for all instances matching the ***session-limits*** qualifier.

DUMP(***instance-qualifier***|ALL|SUMMARYONLY)

Requests formatted dumping of the control information in the resident memory structures. ALL designates all sessions; SUMMARYONLY designates only the anchor.

TERMINATE

Frees all the R4DB2 resident memory structures, terminates all active instances and deletes associated resources. From this point on, all CDC requests will return the inactive status until the next time the `SETACTIVE` command is issued.

When no **`action`** is specified:

Checks if `SETACTIVE` is required in this environment; sets `CC=1` if it is required, and `CC=0` if it was already active. When run with no action, R4DB2CTL will display high level information about all active sessions.

Syntax Elements

The elements used in previous syntax descriptions are as follows:

`session-limits` can be any of the following:

`MAXINSTANCES=n`

Sets the maximum number of concurrently running instances. The default is 32. The enforced maximum is 128, and the enforced minimum is 1.

Once set, changing the max number of instances will terminate the R4DB2 environment, and reactivate with the new value.

`MAXIFIBUFSIZE=n`

Sets the maximum allowed ECSA IFI buffer size (in KB) per instance. The default is 512 (maximum specification is 1,024).

`MAXRESULTSET=n`

Sets the total accumulated size of result rows (containing CDC data), returned from a single SELECT from the user-defined table function.

`MAXIDLEMINS=n`

Sets the number of minutes, after which an inactive instance is eligible for automatic release and reuse. The default is 5 minutes. The maximum is 120 minutes.

`Instance-qualifier`

An identifier designating the Replicate task as it appears in the requestor.

Control Program Completion Codes

- `0` – Normal completion
- `1` – Environment does not exit the ***`session-limits`*** qualifier (when no `action` is specified)

- 4 – Warning
- 8 – Error

IBM DB2 for z/OS Database Source Data Types

The following table shows the IBM DB2 for z/OS database source data types that are supported when using Attunity Replicate and the default mapping from Attunity Replicate data types.

For information on how to view the data type that is mapped in the target, see the section for the target database you are using.

For additional information about Attunity Replicate data types, see Replicate Data Types.

Table 8.25 | IBM DB2 for z/OS Database Source Data Types with Mapping to Attunity Replicate Data Types

IBM DB2 for z/OS Source Data Types	Attunity Replicate Data Types
INTEGER	INT4
SMALLINT	INT2
BIGINT	INT8
DECIMAL (p,s)	NUMERIC (p,s)
	Note If a decimal point is set to a comma (,) in the DB2 configuration, you will need to configure Replicate likewise. For more information, see R4Z Configuration Dependency on z/OS and DB2.
FLOAT (8)	REAL8
DOUBLE	REAL8
REAL	REAL4
DECFLOAT (p)	If precision = 16, then: REAL8 If precision = 34, then: STRING
GRAPHIC	If n<=127, then: WSTRING
VARGRAPHIC	If n<=16k double byte chars, then:

Table 8.25 | IBM DB2 for z/OS Database Source Data Types with Mapping to Attunity Replicate Data Types (Cont.)

IBM DB2 for z/OS Source Data Types	Attunity Replicate Data Types
	WSTRING
LONG VARGRAPHIC	CLOB
CHAR (n)	STRING n<=255
VARCHAR (n)	STRING n<=32k
LONG VARCHAR (n)	CLOB n<=32k
CHAR (n) FOR BIT DATA	BYTES
VARCHAR (n) FOR BIT DATA	CLOB
LONG VARCHAR FOR BIT DATA	BLOB
DATE	DATE
TIME	TIME
TIMESTAMP	DATETIME (6)
BLOB	BLOB
CLOB	CLOB Maximum size: 2 GB
DBCLOB	CLOB Maximum size: 1 G double byte chars
XML	CLOB
BINARY	BYTES
VARBINARY	BYTES
ROWID	IGNORED
TIMESTAMP WITH TIME ZONE	NOT SUPPORTED

Setting General Connection Properties

This section describes how to configure general connection properties. For an explanation of how to configure advanced connection properties, see **Setting Advanced Connection**

Properties below.

Note You can also use IBM DB2 for z/OS File Channel files as a source. For more information, see Using the Attunity Replicate File Channel.

To add an IBM DB2 for z/OS source endpoint to Attunity Replicate:

1. In the Attunity Replicate Console, click **Manage Endpoint Connections** to open the **Manage Endpoints Connections** dialog box. Then click the **New Endpoint Connection** button. For more information on adding an endpoint to Attunity Replicate, see Working with Endpoints.
2. In the **Name** field, type a name for your database. This can be any name that will help to identify the database being used.
3. In the **Description** field, type a description that helps to identify the Microsoft SQL Server database. This is optional.
4. Select **SOURCE** as the database **role**.
5. Select **IBM DB2 for z/OS** as the database **Type**.
6. Choose one of the following:
   - **Use ODBC DSN** - If you choose this option, specify the IBM DB2 for z/OS ODBC DSN.
   - **Use these connection properties** - If you choose this option, enter the IBM DB2 for z/OS **Server** (hostname or IP address), **Port** and **Location** in the designated fields.

     The **Location** should be the DB2 location name defined during the installation. This should be a relational database management system – under z/OS, either a subsystem or a group connection. This is the logical name which serves applications in order to designate resources managed by this system, either using SQL CONNECT instruction, or placing it as a qualifier of a table (preceding the schema name).

     To see the location name, use "-DIS DDF" DB2 command (option 7 under the DB2I panel in ISPF), or look in message DSNL004I in the job log of the <ssid>MSTR address space.
7. Enter the **User name** and **Password** for an authorized user of the specified IBM DB2 for z/OS database. For a list of the permissions that need to be granted to this user, see Required Permissions.

   If you do not know this information, consult with your IBM DB2 for z/OS database administrator (DBA).

Note This information is case sensitive.

Important: Make sure that the specified user has the required access privileges.

8. **Provider:** Leave the default unless it was changed during the driver installation. Note that this should be the same as the name specified in the ODBC Data Source

Administrator.

Setting Advanced Connection Properties

In the **Advanced** tab, you can set advanced connection properties such as overriding CCSID to Character Set mapping, setting Change Data Capture properties, and setting internal Replicate parameters.

Overriding CCSID to Character Set Mapping

In some cases, character data in source tables may be encoded in a different CCSID than what is declared in the source database table definition. For example, a specific table or column definition might indicate that its uses CCSID 500 (EBCDIC International) whereas in fact, it uses CCSID 1148 (ENCDIC International with EURO). In this case, you can tell Replicate that the source definition CCSID 500 should be treated as CCSID 1148 (specifically, the character set named IBM-1148).

Note that when the source table definition specifies CCISD 65535 (meaning character set is unknown), you must specify what character set should be assumed when reading data from that table or column.

Note If there is a conflict between the character set mapping for a specific column and the character set mapping defined in the endpoint settings, the column-level character set mapping takes precedence.

For more information on overriding character set mapping at column level, see Using the Transform Tab.

To do this:

1. In the **Override CCSID to Character Set Mapping** section, click the **New** button.
   A row is added to the table.
2. Enter the CCSID in the **CCSID** column and the code page in the **Character set** column.
3. Repeat to map additional CCSID values.

Converting to a Custom Code Page

Perform the following procedure if your source endpoint tables are defined with an incorrect CCSID and the correct definition is actually in a UCM file.

1. Create or download a mapping data file with the file extension **.ucm**.
2. Create a CNV file for the UCM file by running the following command:
   ```
   <product_dir>\bin\makeconv.exe -v <file_name>.ucm
   ```
   Example:
   ```
   "c:\Program Files\Attunity\Replicate\bin\makeconv.exe" -v 1047_EX.ucm
   ```
   This will create a CNV file with the same name as the UCM file (e.g. **1047_EX.cnv**).

3. Copy the file to the following location:
   `<product_dir>\bin\icudt58l`
4. Add a new character set mapping as follows:
   a. In **CCSID** column, enter the original source CCSID number (e.g. **1047**)
   b. In the **Character set column**, enter the name of the CNV file without the extension (e.g. **1047_EX**).

Change Data Capture Properties

- **Check for changes every:** How often to check for new changes when the database is quiet. When the database is active, changes are captured as soon as they are detected.
- **IFI306 reader stored procedure name:** The name of the Attunity-supplied Stored Procedure, or User-Defined Table Function, which is to access CDC data. Specify the two-part name resulting from the values you have chosen for schema name, &R4ZSCNM and the function name, &R4ZIFITF.
- **IFI306 environment suffix:** Specify the IFI306 environment suffix.
- **IFI306 UDF result set size (MB):** Specify the maximum size to be accumulated by result rows returned by the R4Z-supplied User-Defined Table Function, through CDC tasks using this endpoint. A larger result set will cause less overhead time establishing the IFI session, but will result in greater memory consumption by DB2. Specifically, because the result set is a LOB, you may need to increase the LOBVALA limit in DB2 configuration (DSNZPARM), specifying the maximum LOB size allowed per user. Assuming all CDC enabling tasks are specifying the same ODBC user-ID, to accommodate for all tasks running concurrently, LOBVALA should be set to 2 × 1024 × the following size:

  `SUM[(i=all endpoints) : resultset size in endpoint(i) × #_tasks using endpoint(i)]`

  (Resultset size is in Mbytes, so it needs to be multiplied by 1024 to reflect LOBVALA size, which is in Kbytes).

  Another DSNZPARM limit, LOBVALS, specifies the total size when across all user-IDs; so, in case multiple user IDs are being used, you need to monitor this value as well. Note, however, that LOBVALS is specified in Mbytes, unlike LOBVALA.
- **Automatically enable Data Capture Changes (requires admin privileges):** For Attunity Replicate to be able to capture changes, the Data Capture Changes attribute needs to be set on all relevant source tables. You can either do this manually or allow Replicate to perform this action by leaving this option enabled (the default). When this option is enabled, the connecting user must have ALTER permission on the source tables being captured.

Setting Internal Parameters

Internal parameters are parameters that are not exposed in the UI. With the exception of the parameters listed in Authorized Internal Parameters at the end of this section, you should only use them if instructed by Attunity Support.

To add internal Attunity Replicate parameters:

1. Click the **Internal Parameters** link.
   The **Internal Parameters** dialog box opens.
2. In the edit box, type the name of the parameter you need to add and then click it.
   The parameter is added to the table below the search box with its default value.
3. Change the default value as required.
4. To add more parameters, repeat steps 2 and 3 above.
5. To reset a parameter value to its default, click the "Restore default value" icon at the end of the row.

Authorized Internal Parameters

The following table provides a list of internal parameters tat can be used as required.

Parameter name	Values	Default; Max; Min.
ifi306MessageLevel	0 – No DEBUG traces; 1 – include DEBUG traces; 9 – full DEBUG	0; 0; 9.
ifi306BufferSize	Size (in bytes) of Log data buffer used for IFI reading	65536; 65536; 1048576.
additionalConnectionProperties	<keyword>=<value>;… **Example:** `cursorhold=1;patch2=15`	(empty string)

R4Z Configuration Dependency on z/OS and DB2

This table in this section provides a list of R4Z configuration values, which can be set during installation or as part of the R4Z environment setup, or as part of the endpoint definition.

R4Z Topic	Depends upon Property	Defined at/Inquired by	Resolution
Total Size of R4Z ECSA structures	Available ECSA size	/DNET,CSM,... OWNERID=ALL	Make sure ECSA size is not exceeded.
MAXLOGPSSECS parameter in the R4Z environment	Maximum time DB2 thread waits for a locked resource	DSNZPARM IRLMRWT param. in SDSNSAMP (DSNTIJUZ)	Set at least two seconds below the SSID spec.
MAXRESULSETSIZE-parameter in the R4Z environment	Maximum LOB size available per user-ID	DSNZPARM LOBVALA param. in SDSNSAMP (DSNTIJUZ)	2 * (resultset size * max. session)
Decimal point: period or comma	Designates the decimal editing separator	DSNHDECM DECIMAL= param. In DSNTIJUZ	If the decimal separator is a comma, set the `additionalConnectionProperties` internal parameter to `patch2=15`.

Settings Summary

You can view a summary of your settings by clicking the **Setting Summary** link. This is useful if you need to send a summary of your settings to Attunity Support.

Using Salesforce as a Source

This section describes how to set up and use Salesforce as a source in a replication task.

In this section:

The Salesforce Source Endpoint for Attunity Replicate

The Attunity Replicate Salesforce endpoint uses either a bulk API based on Representational State Transfer (REST) web service for full-load operations or when this is not supported, SOAP web services is used.

The bulk-load API lets you query a large number of records asynchronously by submitting batches that are received in the background by Salesforce.

See Bulk API Limits for information on the limitations on using this API.

In cases where using this API is not supported, SOAP web services is used. In this case, SOAP queries are used to load the Salesforce data into your target endpoint.

Bulk API Limits

The following table describes some of the limitations when using the Bulk API:

Table 8.26 | Bulk API Limitations

Batch Limit	You can submit up to 2,000 batches in a 24-hour period. New batches cannot be created for a job that is more than 24-hours old.
Batch Lifespan	Batch lifespan: Batches and jobs that are older than seven days are removed from the queue regardless of job status. The seven days are measured from the youngest batch associated with a job or the age of the job if there are no batches. New batches cannot be created for a job that is more than 24-hours old.
Batch size	The following are the batch size limitations: » A batch can contain a maximum of: 10,000 records 10,000,000 characters for all the data in the batch

Table 8.26 | Bulk API Limitations (Cont.)

	» A field can contain a maximum of: 32,000 characters » A record can contain a maximum of: 5,000 fields 400,000 characters for all of the fields in the record. » A batch must contain some content. If the batch does not contain any data, an error occurs.
Job/Batch processing time	A five-minute time limit is enforced for processing each 100 records. If the entire batch needs more than 10 minutes to complete processing, the Bulk API places the remainder of the batch in the queue for later processing. If the batch continues to exceed the ten-minute time limit on subsequent attempts, the Bulk API tries to process the batch ten more times. If the batch is not fully processed after ten attempts, the batch is marked as failed. If the batch fails, some records may have been processed successfully. To see which records were processed, see the Salesforce API documentation section called Getting Batch Results.
Job open time	The maximum time that a job can remain open is 24 hours. The Bulk API does not support clients that post only one batch per hour for a long period of time.
Bulk query limitations	Bulk query reduces the number of API requests and scales better. A maximum of ten gigabytes, divided into ten files, can be retrieved by a query. The following are the maximum limits for a bulk query: » Retrieved file size: 1 GB » Number of retrieved files: 10 If the query needs to return more than ten files, the query should be filtered to return less data. Bulk batch sizes are not used for bulk queries. » Number of attempts to query: 10 attempts per 10 minutes If more than 10 attempts are made for the query, the error message `Tried more than ten times` is returned. » Amount of time the results are stored for: 7 days

Working with a Salesforce Account

This section describes the way you work with the Salesforce account when using the Attunity Replicate Salesforce endpoint. It has the following topics:

» Salesforce Account Access

» Using a Salesforce Sandbox

Salesforce Account Access

You or your organization must have a Salesforce account and the Attunity Replicate user must be authorized to access the account.

Using a Salesforce Sandbox

Salesforce Enterprise Edition and Unlimited Edition customers have access to the Salesforce Sandbox, a testing environment that offers a full or partial copy of your organization's live production data.

If you want to use your sandbox environment when working with the Attunity Replicate Salesforce endpoint, configure this in the **Advanced** tab of the **Add Endpoint** dialog box. See Setting Advanced Connection Properties for more information.

Required Permissions (Security Token)

To access a Salesforce endpoint, the user must provide their security token. A security token is an automatically-generated key from Salesforce. When you log in, you must enter the security token as well as your username and password. For information on how to enter your user credentials for the Salesforce account in Attunity Replicate, see Setting General Connection Properties.

Users can obtain a security token by changing their password or resetting their security token in their Salesforce account. When a user changes their password or resets their security token, Salesforce sends a new security token to the email address on the user's Salesforce record. The security token is valid until a user resets their security token, changes their password, or has their password reset.

Limitations

The following limitations apply:

- Tasks configured to use Salesforce as a source endpoint support Full Load replication only.
- The Salesforce source endpoint has limited LOB support. You cannot use an unlimited LOB size for this endpoint. For more information, see Salesforce Data Types.

Salesforce Data Types

The Salesforce endpoint for Attunity Replicate supports most Salesforce data types. The following table shows the Salesforce source data types that are supported when using Attunity Replicate and the default mapping from Attunity Replicate data types.

For information on how to view the data type that is mapped in the target, see the section for the target endpoint you are using.

For additional information about Attunity Replicate data types, see Replicate Data Types.

Table 8.27 | Supported Salesforce Data Types with Mapping to Attunity Replicate Data Types

Salesforce Data Types	Attunity Replicate Data Types
string	STRING
boolean	BOOLEAN
int	INT4
double	NUMERIC
date	DATE
time	TIME
datetime	DATETIME
ID	STRING
reference	STRING
currency	NUMERIC
textarea	STRING
percent	NUMERIC
phone	STRING
url	STRING
email	STRING
combobox	STRING
picklist	STRING
multipicklist	STRING
anyType	STRING
base64	NCLOB
The Salesforce source endpoint has limited LOB support. You cannot use an unlimited LOB size for this endpoint. For more information, see **Task Settings**/Metadata.	

Setting General Connection Properties

This section describes how to configure general connection properties. For an explanation of how to configure advanced connection properties, see **Setting Advanced Connection Properties** below.

To add a Salesforce source endpoint to Attunity Replicate:

1. In the Attunity Replicate console, click the **Manage Endpoint Connections** toolbar button to open the **Manage Endpoints Connections** dialog box. Then click the **New Endpoint Connection** button.
2. In the **Name** field, type a name for your endpoint. This can be any name that will help to identify the endpoint being used.
3. In the **Description** field, type a description that helps to identify the Salesforce account. This is optional.
4. Select **SOURCE** as the endpoint **role**.
5. Select **Salesforce** as the endpoint **Type**.
6. Type the Salesforce authentication information (**User Name**, **Password**) for the Salesforce account you are accessing.

   Note Consider the following:
   - This information is case sensitive.
   - If you want to set custom properties for this endpoint, see Setting Advanced Connection Properties.
   - To determine if you are connected to the endpoint you want to use or if the connection information you entered is correct, click **Test Connection**.

     If the connection is successful a message in green is displayed. If the connection fails, an error message is displayed at the bottom of the dialog box.

     To view the log entry if the connection fails, click **View Log**. The server log is displayed with the information for the connection failure. Note that this button is not available unless the test connection fails.

7. Type the **Security token**. For more information on the Salesforce security token, see Required Permissions (Security Token).

Setting Advanced Connection Properties

In the **Advanced** tab, select one of the following to determine the location of the Salesforce account you are connecting to:

- **Production**: Select this if you are connecting to the default Salesforce endpoint for your account. By default, the connection is to the Salesforce login page, `https://login.salesforce.com`.
- **Sandbox**: Select this if you are connecting to a Salesforce sandbox. This is a test location for a Salesforce account. For more information, see Using a Salesforce Sandbox.
- **Other URL**: Select this connect to a Salesforce account in an alternate location. Type the URL for the location where are working in the adjacent field.

Internal Parameters

Internal parameters are parameters that are not exposed in the UI. You should only use them if instructed by Attunity Support.

To add internal Attunity Replicate parameters:

1. Click the **Internal Parameters** link.

   The **Internal Parameters** dialog box opens.
2. In the edit box, type the name of the parameter you need to add and then click it.
3. The parameter is added to the table below the search box with its default value.
4. Change the default value as required.
5. To reset the parameter value to its default, click the "Restore default value" icon at the end of the row.

Settings Summary

You can view a summary of your settings by clicking the **Setting Summary** link. This is useful if you need to send a summary of your settings to Attunity Support.

Using SAP Application as a Source

This section describes how to define a SAP Application as a source endpoint in a replication task.

In this section:

Prerequisites

The following section describes the prerequisites for working with the Attunity Replicate SAP Application endpoint.

Supported SAP Packages

- Supported SAP backend databases: Oracle, Microsoft SQL Server, and DB2 for LUW.
- Primarily SAP ERP / ECC 6.0 + all EhP levels
- All modules are supported except for HR
- Also support CRM, SRM, GTS and MDG SAP Applications

Set up a Source Endpoint for your SAP Application

Before you can configure the Attunity Replicate SAP endpoint, you first need to configure one of the following source endpoints, according to your SAP Package type:

- Oracle

  See Using Oracle as a Target.
- Microsoft SQL Server

  See Using Microsoft SQL Server as a Target.
- DB2 for LUW

  See Using IBM DB2 for LUW as a Source.

Install the SAP NetWeaver RFC Client

This topic describes how to copy the required the SAP NetWeaver RFC Client files to Attunity Replicate.

- **Windows:** Extract the contents of the **NWRFC_xxx.SAR** file and then copy the **.dll** files from the **nwrfcsdk/lib** directory to the Replicate bin directory.
- **Linux:** Extract the contents of the **NWRFC_xxx.SAR** file and then copy the **.so** files from the **nwrfcsdk/lib** directory to the Replicate lib directory.

Install the Attunity Replicate for SAP Client on the SAP Machine

This section describes how to install the transports that make up the Attunity Replicate for SAP Client.

ECC Systems: For ECC systems, three transports are required: the main transport (InstallCode), a transport with additional logic for ECC SAP systems (InstallCodeECC), and the configuration transport (InstallConfig). Install the code transport first, then the ECC transport, and finally the configuration transport.

Non-ECC systems: For non-ECC systems such as CRM, only two transports are required: the main transport (InstallCode) and the configuration transport (InstallConfig). Install the code transport first, followed by the configuration transport.

Note If you are applying a patch or upgrading the Attunity Replicate for SAP Client, you should only install the main transport and the ECC transport (for ECC systems). Do not install the configuration transport again, or any customizations made to the configuration will be lost.

Permissions Required for Installing Attunity Replicate for SAP Client

Replicate for SAP delivers its own authorization object: `ZR4SAP`. In addition to this authorization object, there are additional authorizations that need to be enabled for the Attunity Replicate software.

SAP Users for Replicate

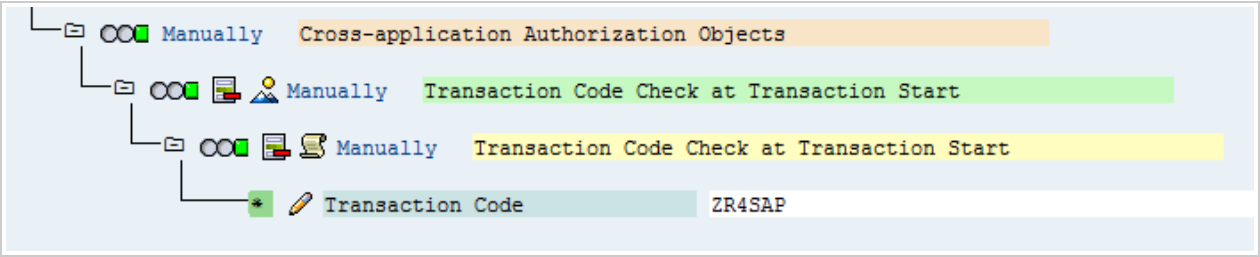
A dialog user in SAP is required to access the Attunity Replicate for SAP Client GUI in SAP. In addition, a communication user is required to support the RFC calls from the Attunity Replicate software to the SAP system.

Identify existing users in SAP or create dedicated users for the Attunity Replicate software.

Authorizations for Replicate

Both the dialog and communication users will need to be assigned to a role with authorization object S_TCODE and value `ZR4SAP`.

Figure 8.3 | Authorization Object S_TCODE with value ZR4SAP:



The communication user will also require the following authorization objects: S_RFC and S_OC_SEND.

Figure 8.4 | Authorization Object S_RFC:

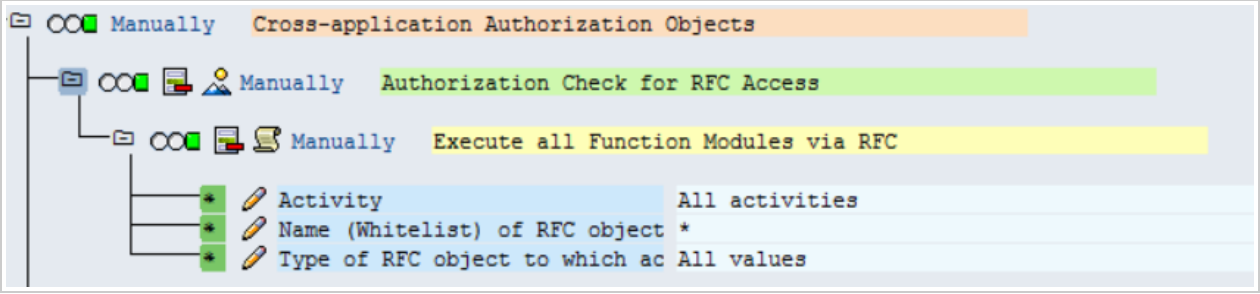
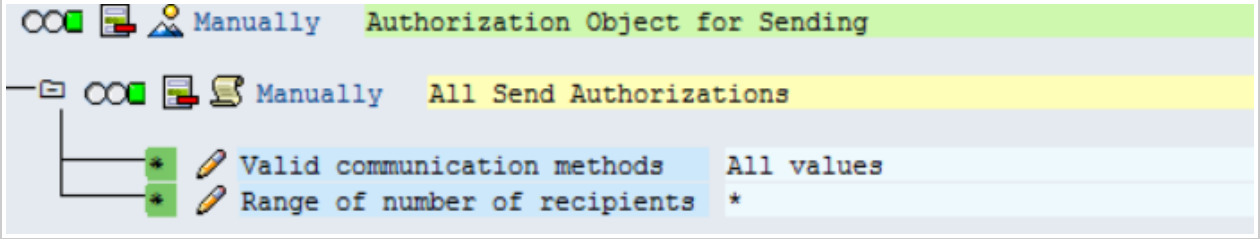


Figure 8.5 | Authorization Object S_OC_SEND:



Importing the Attunity Replicate Transports into the SAP system

There are two types of files required to import the ABAP objects into the SAP system: the data-file and the co-file.

Importing the Data-file

The data-file begins with an "R"

The data-file should be placed in the `/usr/sap/trans/data` file system or in the directory of the server where the transports are stored.

- Typically this is a shared directory, but if not, individually place the file into that directory location for all SAP host servers.
- This file must be accessible by all systems where the Attunity Replicate for SAP Client is to be installed.

- Set the permissions on the file to `All` for the user, `Read and Execute` for the group, and `Read and Execute` for others.
- The owner of the file should be the `<sid>adm` user of the system to be installed. The group ownership should be `sapsys`.

Importing the Co-file

The co-file begins with a "K"

The co-file should be placed in the `/usr/sap/trans/cofiles` file system or in the directory of the server where the transports are stored.

- Typically this is a shared directory, but if not, individually place the file into that directory location for all SAP host servers.
- This file must be accessible by all systems where the Attunity Replicate for SAP Client is to be installed.
- Set the permissions on the file to `All` for the user, `Read and Execute` for the group, and `Read and Execute` for others.
- The owner of the file should be the `<sid>adm` user of the system to be installed. The group ownership should be `sapsys`.

Once the files are in the correct location, import the transport into the system using either the Operating System level transport tools (TP), or the Transport Management System (TMS) internally within SAP.

Importing the Transports via TP

1. Log on to the system at the Operating System level as the <sid> adm.
2. Change the directory to /usr/sap/trans
3. Add the transport to the R/3 buffer with the following command:
   `# tp 'addtobuffer SID'`
4. Import the transport to the target R/3 system with the following command:
   `# tp 'import SID client=000 U16'`

The expected result of the `addtobuffer` step is a successful return code of `0'.

If problems occur during the `addtobuffer` step, it is likely there is a problem with the files. They may be missing, in the wrong location, or have incorrect ownership or permissions.

The expected result of the `import` step is a successful return code of either `0' or `4'. A return code of `8', `12' or `16' indicates transport failure. Return codes higher than `16' indicate a major failure within the transport tool. If this occurs, check the present working directory to ensure the correct location. Also, check the files for existence, location, and proper ownership and access.

If problems exist during the import, retry the import step. If problems persist, check the import and activation logs for failure reason. These files are in the `/usr/sap/trans/log` location and named.`R6U` (the `?' stands in as a wildcard).

Importing the Transports via TMS

Beginning in R/3 version 4.0, SAP allows importing transports through the SAP system via transaction code STMS.

Note Security authorization in the SAP system must include proper access to import the transport request.

1. Sign-on to SAP system with a User ID that contains proper authority.
2. Execute transaction **STMS**.
3. Select the **Transport** (Truck) icon from the toolbar.
4. Select the desired system for import.
5. Add the Transport to the import queue by selecting the following path from the menu list:

   Extras > **Other Requests** > **Add**

   Add the transport request number to the proper import queue and execute. Re-authentication of the user.s SAP User ID is likely in order to complete the step.

   If an Information message is received that the "Transport request is invalid" check that the transport number was typed correctly. Otherwise, it may indicate a problem with the files. Verification of existence, location, permissions, or ownership may be needed.
6. Import the Transport request by selecting the transport number from the queue, and clicking the **Import** (Truck) icon from the toolbar. Set the target client to either `000' or any other valid client within the system and execute with the truck icon. Once again, re-authentication of the SAP User ID may be necessary. The transport will execute in asynchronous mode; a record of success or failure can be found in the transport logs.
7. The system will return to the import queue screen, where the Transport results can be checked. Select the **Logs** icon from the toolbar, or follow the menu path:

   Request > **Display** > **Logs**

   Locate the source system and verify all relevant logs. For this transport there should be 5 logs:

   - **DD Import**
   - **DD Activation**
   - **Import**
   - **Check Versions**
   - **ABAP/scrn. Generation**

   All logs should display `Ended OK` (return code 0) or `Ended with warning` (return code 4). If any logs are missing, or display a return code of 8 or higher, follow the instructions in step 6 to re-import the transport.

Upgrading, Patching and Uninstalling the Attunity Replicate SAP Client

The "Install Main" transport should be applied if you need to upgrade or patch Attunity Replicate SAP Client.

The "Delete Main" and "Delete Package" transports should be applied if you need to completely remove Attunity Replicate SAP Client from your SAP system.

To update or patch the Attunity Replicate SAP Client

- Apply the new "Install Main" transport.

To uninstall the Attunity Replicate SAP Client

- Apply the "Delete Main" transport, followed by the "Delete Package" transport.

Note Do not uninstall the Replicate for SAP Client if you are also running Attunity Gold Client, as this will uninstall some components that are shared by both products.

Managing Business Groups and Tables

This prerequisite is only necessary if you want to edit the default Business Groups and/or tables before replicating them to the target endpoint.

Before you can manage business groups and tables, you first need to launch the SAP Client UI.

To launch the SAP Client UI:

1. Open your SAP client console.
2. Double-click one of the SAP Application Sources.
   You will be prompted for your user name and password.
3. Enter your credentials for logging in to the selected SAP Application Source.
4. Enter `/nzr4sap` in the drop-down list at the top of the console and then press [Enter].
5. Click the **Business Groups Configuration** button.
   A list of Business Groups is displayed in the left pane.

Managing Business Groups

To add a new Business Group

1. Click the **Create** toolbar button.
   The **Create Business Group** dialog box opens.
2. Enter a name for your Business Group and then press [Enter].
   The new Business Group is added to the **Business Groups** list in the left pane.

To duplicate a Business Group

1. Click the **Copy** toolbar button.
   The **Business Group Configuration** dialog box opens.
2. In the **New Bus Object** field, enter a name for the new Business Group and then press [Enter].
   The duplicated Business Group is added to the **Business Groups** list in the left pane.

To delete a Business Group

» Select the Business Group you want to delete and then click the **Delete** toolbar button.
The Business Group is deleted.

Managing Tables

To add a new table to a Business Group

1. In the left pane, expand the desired Business Group.
2. Double-click the **Tables** icon.
   A list of tables is shown in the right pane.
3. Click the  button above the table list to enter Edit mode.
4. Click the  button that appears to the right of the  button.
   An empty row is added to the tables list.
5. Enter the **Table Name** (i.e. the virtual ABAP table) and the name of the corresponding **Source Table** (i.e. the physical table).
6. To save your changes click the **Save** button in the main toolbar.

To remove a table from a Business Group

1. In the left pane, expand the desired Business Group.
2. Double-click the **Tables** icon.
   A list of tables is shown in the right pane.
3. Click the  button above the table list to enter Edit mode.
4. Select the table you want to delete.
5. Click the  button that appears to the right of the  button.
   The table is deleted.
6. To save your changes click the **Save** button in the main toolbar.

Target Collation

As SAP is case-sensitive, when a Replicate task is defined with a SAP Application source, the target endpoints need to be set up with case-sensitive collation.

Limitations

When using SAP Application as a source endpoint in a Replicate task, the following limitations apply:

- A task with a SAP Application source and a File Channel target may replicate some tables twice - the requested table and the underlying table. To prevent this from happening, exclude the underlying table from the list of tables to be replicated.
- When a task is defined with a SAP Application source, the **Applied Changes Details** monitoring metrics in the Change Processing tab may be incorrect for clustered and pooled tables.

SAP Application Source Data Types

The SAP Application endpoint for Attunity Replicate supports most SAP data types. The table below shows the SAP source data types that are supported when using Attunity Replicate and the default mapping to the Attunity Replicate data types.

For information on how the data type is mapped to the target, see the chapter for the target endpoint you are using.

For additional information about Attunity Replicate data types, see Replicate Data Types.

Table 8.28 | Supported SAP Data Types with Mapping to Attunity Replicate Data Types

ABAB Type	ABAB Type Description	SAP Type	Attunity Replicate Data Type
h	Table type		BYTES
V	Character string (old Dictionary type VARC)		STRING
C	Character string		STRING
N	Character string with only digits		STRING
D	Date (string: YYYYMMDD)		DATE
T	Time (string: HHMMSS)		TIME
X	Byte sequence	INT4 (4-byte integer)	I4

Table 8.28 | Supported SAP Data Types with Mapping to Attunity Replicate Data Types (Cont.)

ABAB Type	ABAB Type Description	SAP Type	Attunity Replicate Data Type
		INT2 (2-byte integer)	I2
		INT1 (1-byte integer)	I1
		ELSE	If backend type is NUMERIC: NUMERIC If length = 0: BLOB If length > 0: BYTES
I	Integer number (4-byte integer with sign)		INT4
b	2-byte integer		INT2
s	1-byte integer		INT1
P	Packed number		NUMERIC
F	Floating point number to accuracy of 8 bytes		R8
g	Character string with variable length		STRING
y	Byte sequence with variable length		BLOB BYTES
u	Structured type, flat		BYTES
v	Structured type, deep		BYTES
r	Reference to class/interface		BYTES
i	Reference to data object		BYTES

Setting General Connection Properties

This section describes how to set up connection parameters for a specific SAP Application server or for an SAP system using load balancing.

To connect to a specific SAP Application server

1. In the Attunity Replicate console, click the **Manage Endpoint Connections** toolbar button to open the **Manage Endpoints Connections** dialog box. Then click the **New Endpoint Connection** button.
2. In the **Name** field, enter a display name for your endpoint.
3. Optionally, in the **Description** field, enter a description for the SAP Application endpoint.
4. Select **Source** as the database **role**.
5. Select **SAP Application** as the database **Type**.
6. From the **Connection type** drop-down list, select **Type A**.
7. In the **Server name** field, enter the IP address of the Application Server on which the SAP Application source is located.
8. In the **Instance number** field, enter the instance number of the SAP Application source you want to replicate.
9. In the **Client** field, enter the System ID of the SAP Application source you want to replicate.
10. Enter your credentials (**User Name**, **Password**) for accessing the SAP Application source.

    Note: These are the credentials for the communication user created earlier in SAP.
11. In the **Backend database** field, click the Browse button and then select the name of the Attunity Replicate endpoint you configured earlier. See also Set up a Source Endpoint for your SAP Application.

To connect to an SAP system using load balancing

1. In the Attunity Replicate console, click the **Manage Endpoint Connections** toolbar button to open the **Manage EndpointsConnections** dialog box.
2. In the **Name** field, enter a display name for your endpoint.
3. Optionally, in the **Description** field, enter a description for the SAP Application endpoint.
4. Select **Source** as the database **role**.
5. Select **SAP Application** as the database **Type**.
6. From the **Connection type** drop-down list, select **Type B**.
7. In the **Message server** field, enter the host name or IP address of the message server host.
8. In the **Application servers group name** field, enter the name of the SAP server group. This is an optional group of application servers in a load balancing connection.
9. In the **SAP system name** field, enter the SAP R/3 name.
10. In the **Message server service** field, enter the name of the SAP message server service as specified in the following file:

    `<system drive>:\WINDOWS\system32\drivers\etc\services`

If you do not specify a value, the Data Provider for SAP uses the following default name:
`sapms<R/3 system name>`

11. In the **Client** field, enter the System ID of the SAP Application source you want to replicate.
12. Enter your credentials (**User Name**, **Password**) for accessing the SAP Application source.

    Note: These are the credentials for the communication user created earlier in SAP.
13. In the **Backend database** field, click the Browse button and then select the name of the Attunity Replicate endpoint you configured earlier. See also Set up a Source Endpoint for your SAP Application.

Setting Advanced Properties

In the **Advanced** tab, you can set the following parameters:

- **RFC call batch**: The number of concurrent RFC calls made from Replicate back to the SAP system. If you encounter performance issues, increasing this number may help, but may also adversely affect monitoring updates.

Internal Parameters

Internal parameters are parameters that are not exposed in the UI and should only be used if instructed by Attunity Support.

To add internal Attunity Replicate parameters:

1. Click the **Internal Parameters** link.

   The **Internal Parameters** dialog box opens.
2. In the edit box, type the name of the parameter you need to add and then click it.
3. The parameter is added to the table below the search box with its default value.
4. Change the default value as required.
5. To reset the parameter value to its default, click the "Restore default value" icon at the end of the row.

Settings Summary

You can view a summary of your setting by clicking the **View Setting Summary** link. This is useful if you need to send a summary of your setting to Attunity Support.

Using ODBC to Connect to a Source

This section describes how to use ODBC connectivity to connect to a source endpoint.

In this section:

Prerequisites

The following section describes the prerequisites for working with Attunity Replicate and an ODBC endpoint.

Attunity Replicate Server for Windows

You can connect an endpoint to Attunity Replicate using ODBC by indicating the DSN (Data Source Name). In this case you must be sure that a DSN is defined for the ODBC endpoint on the computer where Attunity Replicate is installed.

1. Install an endpoint client on the computer where Attunity Replicate is installed. The client you install depends on the ODBC provider you are using. For example, if you are using an IBM DB2 endpoint, install an IBM DB2 client.

   Note You must use a 64-bit ODBC provider client to work with Attunity Replicate.

2. Use the ODBC Data Source Administrator to create a System DSN. The Data Source is located in the Windows control panel.

Attunity Replicate Server for Linux

The following section describes the steps you need to perform to work with Attunity Replicate for Linux and ODBC as a source or target endpoint in a Attunity Replicate task.

1. On the Attunity Replicate Server machine, install the ODBC client that you want to use (e.g. postgreSQL).
2. Makes sure that the `/etc/odbcinst.ini` file contains the correct entry for the driver you installed, as in the following example:

```
[PostgeSQL]
Description = ODBC for PostgreSQL
Driver = /usr/lib/psqlodbc.so
Setup = /usr/lib/libodbcpsqlS.so
Driver64 = /usr/lib64/psqlodbc.so
```

```
Setup64 = /usr/lib64/libodbcpsqlS.so
FileUsage = 1
```

Note To access an IBM DB2 for LUW target using ODBC, make sure that you specify the **libdb2o.so** driver (and not **libdb2.so**).

3. Define a DSN for the installed driver by editing the `/etc/odbc.ini` file, as in the following example:

```
[Postgre_DSN]
Description = Test
Driver = /usr/lib64/psqlodbc.so
Endpoint = MyDatabase
Servername = 12.3.45.678
Port = 5432
```

Limitations

When using ODBC as a source, the following limitations apply:

- UPDATES to primary key fields are not supported. To update the field, define it as a unique index instead.
- The ODBC Source endpoint supports full-load operations only.
- For providers that do not support batch operations, you must set the `RowByRow=true` internal parameter according to the description provided in Internal Parameters.

ODBC Source Data Types

The following table shows the ODBC source data types that are supported when using Attunity Replicate and the default mapping from Attunity Replicate data types.

For information on how to view the data type that is mapped in the target, see the section for the target endpoint you are using.

For additional information about Attunity Replicate data types, see Replicate Data Types.

Table 8.29 | Supported ODBC Source Data Types with Mapping to Attunity Replicate Data Types

ODBC Data Types	Attunity Replicate Data Types
SQL_BIT	**BOOLEAN**
SQL_TINYINT	**INT1**
	UINT1

Table 8.29 | Supported ODBC Source Data Types with Mapping to Attunity Replicate Data Types (Cont.)

ODBC Data Types	Attunity Replicate **Data Types**
	Note SQL data types are mapped to unsigned data types when the `UNSIGNED_ATTRIBUTE` is set to `SQL_TRUE` for the data type being mapped.
SQL_SMALLINT	**INT2** **UINT2** **Note** SQL data types are mapped to unsigned data types when the `UNSIGNED_ATTRIBUTE` is set to `SQL_TRUE` for the data type being mapped.
SQL_INTEGER	**INT4** **UINT4** **Note** SQL data types are mapped to unsigned data types when the `UNSIGNED_ATTRIBUTE` is set to `SQL_TRUE` for the data type being mapped.
SQL_BIGINT	**INT8** **UINT8** **Note** SQL data types are mapped to unsigned data types when the `UNSIGNED_ATTRIBUTE` is set to `SQL_TRUE` for the data type being mapped.
SQL_DOUBLE	**REAL8**
SQL_FLOAT	**REAL8**
SQL_REAL	**REAL8**
SQL_NUMERIC (P,S)	**NUMERIC (P,S)** **REAL8** The SQL_NUMERIC data type is mapped to REAL8 when *at least one* of the following is true: » Precision > 38

Table 8.29 | Supported ODBC Source Data Types with Mapping to Attunity Replicate Data Types (Cont.)

ODBC Data Types	Attunity Replicate **Data Types**
	» Scale < 0 » Scale > 38 » Scale > Precision
SQL_DECIMAL (P,S)	**NUMERIC (P,S)** **REAL 8** The SQL_NUMERIC data type is mapped to REAL8 when *at least one* of the following is true: » Precision > 38 » Scale < 0 » Scale > 38 » Scale > Precision
SQL_DATE SQL_TYPE_DATE	**DATE**
SQL_TIME SQL_TYPE_TIME	**TIME**
SQL_TIMESTAMP SQL_TYPE_TIMESTAMP	**DATETIME**
SQL_CHAR SQL_VARCHAR	**STRING**
SQL_WCHAR SQL_WVARCHAR	**WSTRING**
SQL_LONGVARCHAR	**CLOB** **Note** To use this data type with Attunity Replicate, you must enable the use of CLOBs for a specific task. During CDC, CLOB data types are supported only in tables that include a primary key. For more information, see **LOB support** in **Task Settings**/Metadata.

Table 8.29 | Supported ODBC Source Data Types with Mapping to Attunity Replicate Data Types (Cont.)

ODBC Data Types	Attunity Replicate Data Types
SQL_WLONGVARCHAR	**NCLOB**
Note To use this data type with Attunity Replicate, you must enable the use of NCBLOBs for a specific task. During CDC, NCLOB data types are supported only in tables that include a primary key. For more information, see **LOB support** in **Task Settings**/Metadata.	
SQL_BINARY	**BYTES**
SQL_LONGVARBINARY	**BLOB**
Note To use this data type with Attunity Replicate, you must enable the use of BLOBs for a specific task. BLOB data types are supported only in tables that include a primary key. For more information, see **LOB support** in **Task Settings**/Metadata.	
SQL_GUID	**STRING**
SQL_INTERVAL_YEAR	**STRING**w
SQL_INTERVAL_MONTH	
SQL_INTERVAL_DAY	
SQL_INTERVAL_MINUTE	
SQL_INTERVAL_HOUR	
SQL_INTERVAL_SECOND	
SQL_INTERVAL_YEAR_TO_MONTH	
SQL_INTERVAL_DAY_TO_HOUR	

Table 8.29 | Supported ODBC Source Data Types with Mapping to Attunity Replicate Data Types (Cont.)

ODBC Data Types	Attunity Replicate Data Types
SQL_INTERVAL_DAY_TO_MINUTE SQL_INTERVAL_DAY_TO_SECOND SQL_INTERVAL_HOUR_TO_MINUTE SQL_INTERVAL_HOUR_TO_SECOND SQL_INTERVAL_MINUTE_TO_SECOND	
Provider specific data types **Note** If column length is 0 or > 4000 then: To use this data type with Attunity Replicate, you must enable the use of BLOBs for a specific task. BLOB data types are supported only in tables that include a primary key. For more information, see **LOB support** in **Task Settings**/Metadata.	If column length is < or = 4000: **BYTES** If column length is 0 or > 4000: **BLOB**

Setting General Connection Properties

This section describes how to configure general connection properties. For an explanation of how to configure advanced connection properties, see **Setting Advanced Connection Properties** below.

To add an ODBC source endpoint to Attunity Replicate:

1. In the Attunity Replicate Console, click **Manage Endpoint Connections** to open the **Manage Endpoints Connections** dialog box. Then click the **New Endpoint Connection** button.
2. In the **Name** field, type a name for your ODBC endpoint. This can be any name that will help to identify the endpoint being used.
3. In the **Description** field, type a description that helps to identify the ODBC endpoint. This is optional.
4. Select **SOURCE** as the endpoint **role**.
5. Select **ODBC** as the endpoint **Type**.
6. Select *one* of the following:

- If the DSN you want to use is not included in the list, make sure that the endpoint client is installed on the computer with Attunity Replicate and that the DSN is defined. Note that the ODBC provider client must be 64-bit. For more information, see Prerequisites .

  Note If you are using an ARC CDC Agent as the source in a Attunity Replicate task, you cannot select the DSN for the Attunity ODBC driver as the target. In this case, to use Attunity ODBC as a source, you must enter the connection string manually by selecting **Connection String** and following the directions for that option in this procedure.

- **Connection String**: Select this to connect to an ODBC-supported endpoint using a connection string then type a valid connection string in the field below. For information on how to create a connection string, see the documentation for the ODBC endpoint provider you are using.

  Note that if you specify a password in your connection string, it will be revealed as plain text in the task log files. It is therefore recommended to specify the password in the GUI **Password** field.

  Note

  - You can use the **Advanced** tab to add specific properties and create a custom connect string. In this case, you do not need to enter information in this tab. For more information on using the **Advanced** tab, see Setting Advanced Connection Properties.
  - To determine if you are connected to the endpoint you want to use or if the connection information you entered is correct, click **Test Connection**.

    If the connection is successful a message in green is displayed. If the connection fails, an error message is displayed at the bottom of the dialog box.

    To view the log entry if the connection fails, click **View Log**. The server log is displayed with the information for the connection failure. Note that this button is not available unless the test connection fails.

7. Type the authentication information (**User Name**, **Password**) for the authorized user for the ODBC endpoint being used. For example, the IBM DB2 system administrator if you are using a IBM DB2 provider. If you do not know this information, see your ODBC Endpoint System Administrator.

   Note

   - When you select **Connection String** be sure to include **User name/password** information in the connection string that you type in the box.

If you are using the **Advanced** tab to create a custom string, make sure to include the **User Name** and **Password** properties. For more information, see Setting Advanced Connection Properties.

- This information is case sensitive.
- You can set custom properties in the **Advanced** tab. For more information, see Setting Advanced Connection Properties.

Important: Make sure that the ODBC endpoint user has the correct access privileges for the ODBC provider being used.

Setting Advanced Connection Properties

In the **Advanced** tab, you can set the following properties:

- **Provider syntax**: Select the name of the provider syntax if you are using an alternate provider syntax.

Internal Parameters

Internal parameters are parameters that are not exposed in the UI. You should only use them if instructed by Attunity Support.

To add internal Attunity Replicate parameters:

1. Click the **Internal Parameters** link.
   The **Internal Parameters** dialog box opens.
2. In the edit box, type the name of the parameter you need to add and then click it.
3. The parameter is added to the table below the search box with its default value.
4. Change the default value as required.
5. To reset the parameter value to its default, click the "Restore default value" icon at the end of the row.

Settings Summary

You can view a summary of your settings by clicking the **Setting Summary** link. This is useful if you need to send a summary of your settings to Attunity Support.

Using ARC CDC Solutions in Attunity Replicate

This section describes how to use an ARC (Attunity Replicate Connect) CDC Solution as an Attunity Replicate endpoint.

Note For all ARC sources, it is strongly recommended to install the ARC Agent in the same data center as the Replicate server.

In this section:

Prerequisites for Using ARC CDC Solutions

To use an ARC CDC Solution, you must have the following installed somewhere in your network in addition to the database you are working with.

- **ARC version 5.5 or above:** This must be installed on the same computer as the database you are using. You will need the installation kit for the computer platform that your database runs on. For example, if you are using a an IBM DB2 database for z/OS, install ARC on the same mainframe computer where your IBM DB2 database is located. For information on how to install ARC, see the *Attunity Integration Suite Installation Guide* for the computer platform that is relevant to the CDC Solution you are working with.
- **Attunity Studio version 5.3.2 or above:** Attunity Studio is used to set up a CDC Solution. This will create the CDC Solution that can be used in the Attunity Replicate replication task. Attunity Studio must be installed on a Windows computer. For information on installing Attunity Studio, see the *Attunity Studio Installation Guide*.
- **When the ARC database is on DB400-AS4002**: To apply deletes to the target, journaling must be set to *`BOTH`.
- **ARC relational data sources that support table ownership:** If the table owner contains an underscore, you must create the ARC solution with a default table owner.
- **When the source endpoint is IBM IMS (ARC):** The ARC IMS Bulk data source is always created as IMS-DLI. You should specify the correct ARC IMS **Bulk started task** in the endpoint settings. The ARC USERLIB library contains the following started task examples:

- NVIMSSRV for IMS DLI access
- NVBMPSRV for IMS BMP access

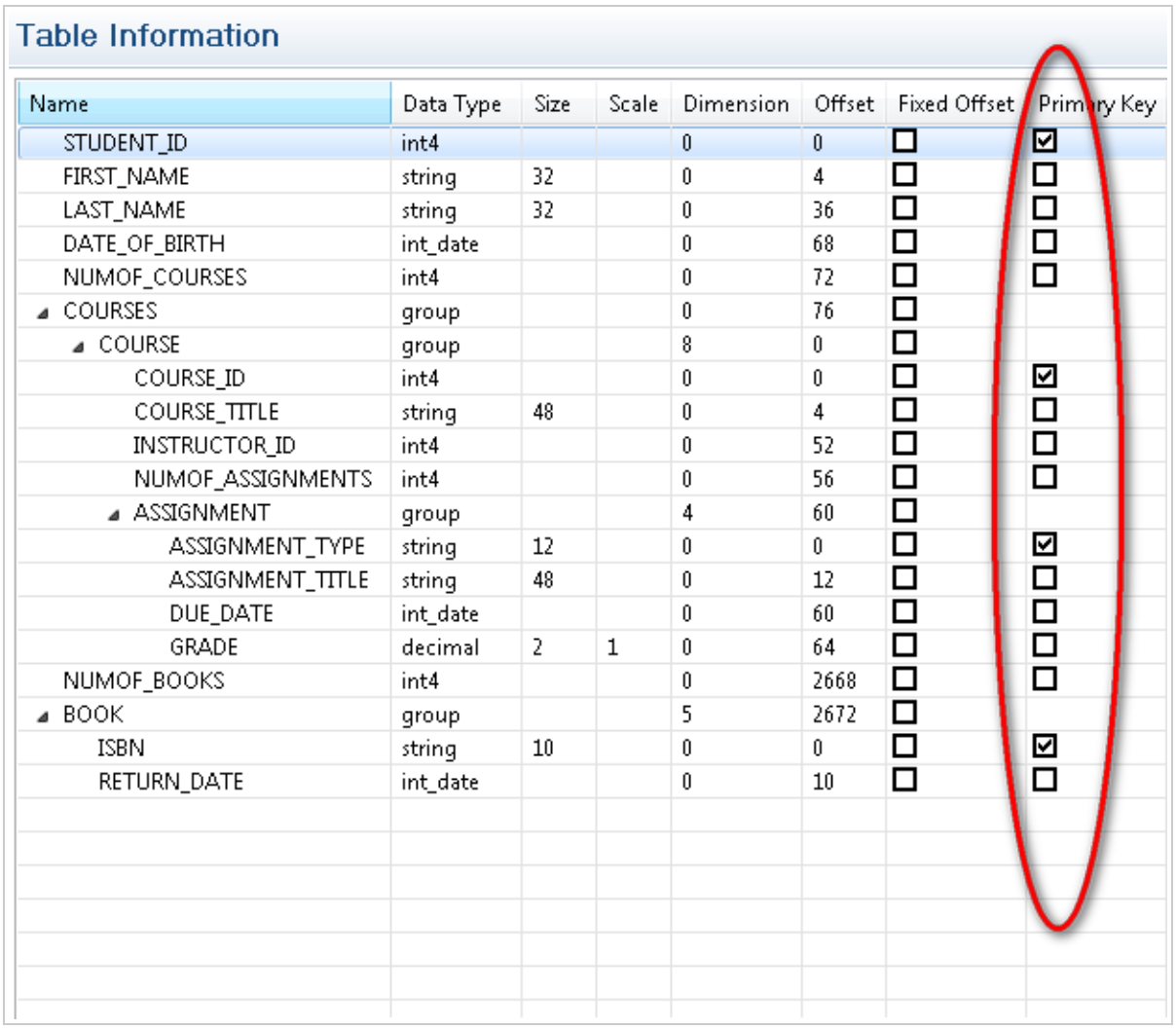
For more information on creating ARC Solutions, please refer to the *Attunity Replicate Connect User Guide and Reference*.

Note For information about installing the database you are working with, see the installation guide for that database.

Additional Prerequisites when Using ARC Non-Relational Sources

When using any of the Using ARC CDC Agents as Endpoints, the following prerequisites also apply:

- ARC 5.5 or above installed on the Attunity Replicate machine.
- If the source tables contain Primary Keys, edit the source table metadata in Attunity Studio and mark the Primary Key columns as shown in the figure below. This should be done at the start of creating the CDC solution when importing the tables/files.



For more information on selecting Primary Keys in Attunity Studio, refer to the "Working with Metadata in Attunity Studio" chapter in the *ARC User Guide and Reference*.

ARC CDC Solution Security Considerations

For an explanation of the security configurations and permissions necessary, see the CDC Solution reference in the *Attunity Replicate Connect User Guide and Reference*.

Encrypting Communications Between Replicate and ARC Data Sources

You can encrypt sessions between Replicate and ARC data sources. When a session is encrypted, all communications between Replicate and the selected ARC data source will be encrypted using AES-256 bit encryption. When capturing changes from a relational data source, the encryption key needs to be defined in two locations: The Attunity Replicate ARC database and the ARC Agent machine. However, when capturing changes from a *non*-relational database, the encryption key needs to be defined in *four* different locations: The

Attunity Replicate ARC database, the ARC Agent machine, the ARC Router machine, and the Router Authenticator.

To encrypt communications between Replicate and ARC data sources:

1. On the **Agent** machine, create an encryption key as follows:
   a. Open Attunity Studio in **Design** view.
   b. In the **Configuration** tab, expand the machine on which your ARC Solution's Agent is installed.
   c. Expand the **Users** folder and select **NAV**.

      The **User: NAV** tab opens.
   d. To the right of the **Encryption Keys** list (in the lower half of the screen), click the **Add** button.

      The **Encryption Key** dialog opens.
   e. Enter an encryption key name and value and then click **OK**.

   Note Steps 2-4 apply to *non-relational* ARC data sources only (e.g. VSAM). If you are working with a relational ARC data source, continue to Step 5.

2. On the **Router** machine, create an encryption key which has the same values as the encryption key that you created on the **Agent** machine. The procedure is the same as described in **Step 1**, but instead of expanding the machine on which your ARC Solution's Agent is installed, expand the machine on which your ARC Solution's Router is installed.
3. On the **Router** machine, define the Agent as an authenticator according to the following steps:
   a. In the **Configuration** tab, expand the machine on which the Router is installed. Then, right-click your solution's Router binding (e.g vsam_router) and select **Open**.
   b. In the **Machines** tab, click the **Security** button.

      The **NAV** tab opens.
   c. To the right of the **Authenticators** list, click the **Add** button.

      The **Add Authenticator** dialog box opens.
   d. From the **Resource type**, drop-down list, select **Adapter**.
   e. In the **Resource name** field, specify the name of your solution's Agent as it appears under the **Adapters** folder (e.g VSAM_ag).
   f. At the bottom of the dialog box, select the **Encryption key** check box and then specify the encryption key name and value in the designated fields. These values must be the same as the encryption key values defined in **Step 1**.
4. In the Router's **Properties** tab, expand the **comm** property and set the **defaultEncryptionMethod** property to **AES**.

Note If the **Properties** tab is not displayed, open the **Preferences** dialog box (by selecting **Preferences** from the **Windows** menu), navigate to **Studio** and then select the **Show advanced environment parameters** option in the **Advanced** tab.

5. In the **Advanced** tab of the Replicate ARC database, specify the encryption key name and value. These values must be the same as the encryption key values defined in **Step 1**.

   For more information on the Advanced tab, see Using ARC CDC Agents as Endpoints.

   See also: Using ARC CDC Agents as Endpoints.

Limitations

When working with ARC data sources, the following limitations apply:

- **IBM DB2 on iSeries (ARC):** Table and field names that contain the "/" character are not supported.
- **IBM DB2 on z/OS (ARC):** LOB columns are not supported.
- Only one Replicate task can work with the same ARC Agent concurrently.
- Replication of DDL changes to the target endpoint is not supported.

ARC Data Types

This section describes the mapping of ARC data types to Attunity Replicate data types. It has the following sections:

- ARC Source Data Type Mapping

For an explanation of the supported data types for the ARC CDC Solution you are using, see the CDC Solution reference in the *Attunity Replicate Connect User Guide and Reference*.

ARC Source Data Type Mapping

The following table shows the ARC source data types that are supported when using Attunity Replicate and the default mapping to Attunity Replicate data types.

For information on how to view the data type that is mapped in the target, see the section for the target database you are using.

For additional information about Attunity Replicate data types, see Replicate Data Types.

Table 8.30 | Supported ARC Data Types with Mapping to Attunity Replicate Data Types

ARC Data Types	Attunity Replicate Data Types
ARC SQL Server	

Table 8.30 | Supported ARC Data Types with Mapping to Attunity Replicate Data Types (Cont.)

ARC Data Types	Attunity Replicate Data Types
INT	**REAL4**
REAL	**REAL4**
FLOAT	**REAL8**
BIT	**INT1**
TINYINT	**INT1**
SMALLINT	**INT2**
BIGINT	**NUMERIC**
DECIMAL	**NUMERIC**
NUMERIC	**NUMERIC**
MONEY	**NUMERIC**
SMALLMONEY	**NUMERIC**
DATETIME	**DATETIME**
SMALLDATETIME	**DATETIME**
CHAR	**STRING**
VARCHAR	**STRING**
NCHAR	**STRING**
NVARCHAR	**STRING**
BINARY	**BYTES**
VARBINARY	**BYTES**
TIMESTAMP	**BYTES**
UNIQUEIDENTIFER	**STRING**
ARC DB2 iSeries	
SMALLINT	**INT2**
INTEGER	**INT4**
BIGINT	**NUMERIC**
DECIMAL	**NUMERIC**
NUMERIC	**NUMERIC**

Table 8.30 | Supported ARC Data Types with Mapping to Attunity Replicate Data Types (Cont.)

ARC Data Types	Attunity Replicate Data Types
FLOAT	**REAL4**
DOUBLE	**REAL8**
DATE	**DATE**
TIME	**TIME**
TIMESTAMP	**DATETIME**
CHAR	**STRING**
VARCHAR	**STRING**
GRAPHIC	**STRING**
VARG	**STRING**
DATALINK	**STRING**
ROWID	**STRING**
BINARY	**BYTES**
ARC DB2 z/OS	
(VAR) CHAR	**STRING**
(VAR) CHAR for bit data	**BIT**
VARCHAR	**STRING**
BIGINT	**NUMERIC**
DECIMAL	**NUMERIC**
SMALLINT	**INT2**
INTEGER	**INT4**
FLOAT	**REAL8**
DOUBLE	**REAL8**
DATE	**DATE**
TIME	**TIME**
TIMESTAMP (n)	**DATETIME (6)**
GRAPHIC	**BYTES**
CLOB	**CLOB**
BLOB	**BLOB**

Working with ARC CDC Solutions

To use a CDC Solution from the Attunity Integration Suite, you must first create a CDC Solution in Attunity Studio. Then create a new database using the CDC Solution you created as the Attunity Replicate database. You can then use this database as your source for any task that you create. To use ARC CDC Solutions, carry out the following:

- Create an ARC CDC Solution in Attunity Studio
- Add the ARC Data Source to Attunity Replicate
- Add the ARC CDC Solution Endpoint to a Task

Create an ARC CDC Solution in Attunity Studio

Before you can begin to work with an ARC CDC Solution in Attunity Replicate, you must create a CDC solution using one of the supported ARC CDC Solutions using Attunity Studio. For information on the required ARC installation necessary to create a CDC solution, see Prerequisites for Using ARC CDC Solutions.

To create a CDC solution in Attunity Studio:

1. Using Attunity Studio, create a CDC Solution using the CDC Solution that you want to use as your source database in Attunity Replicate.

   For information on creating a CDC Solution, refer to the *Attunity Integration Suite User Guide and Reference*.
2. At the end of the process for creating a CDC solution, you must deploy the solution. Do *not* activate the solution. Attunity Replicate activates the solution automatically when you begin to work with the CDC Solution.

   Note If you activate the solution, then disable the router and staging area workspaces and keep the agent workspace enabled. For more information, see the *Attunity Integration Suite User Guide and Reference*.

Add the ARC Data Source to Attunity Replicate

The next step is to add the ARC Data Source to Attunity Replicate. You do this by adding a database and selecting one of the supported ARC database types.

If you selected one of the supported relational data sources, continue from Adding a Relational ARC Data Source to Attunity Replicate.

If you selected one of the supported non-relational data sources, continue from Adding a Non-Relational ARC Data Source to Attunity Replicate. See also Additional Prerequisites when Using ARC Non-Relational Sources.

For information on how to add endpoints, see Working with Endpoints.

Adding a Relational ARC Data Source to Attunity Replicate

To add a relational ARC data source to Attunity Replicate:

1. In the Attunity Replicate console, click **Manage Endpoint Connections** to open the **Manage Endpoint Connections** dialog box and then click **New Endpoint Connection**. For more information on adding an endpoint to Attunity Replicate, see Working with Endpoints.
2. In the **Name** field, type a name for your database. This can be any name that will help to identify the database being used.
3. In the **Description** field, type a description that helps to identify the ARC CDC Solution. This is optional.
4. Select **Source** as the **role**.
5. Select a relational ARC data source from the **Type** list. The ARC data sources are listed as <*Data Source*> **(ARC)**, for example **IBM DB2 on z/OS (ARC)**. For a list of supported relational data sources, see Using ARC CDC Agents as Endpoints.
6. In the **Host/IP** field, type the name or IP Address of the computer where the CDC Solution (data source) you defined in Attunity Studio is located.
7. In the **Port** field, type the port number for the port you used when creating the CDC Solution in Attunity Studio. The default port number is 2551.
8. In the **CDC Solution** field, enter the name of the solution you defined when you created the data source in Attunity Studio.
9. In the **User name** and **Password** fields, enter the username and password required to access the database.
10. Click **OK** to add the database to Attunity Replicate. You can use this database as the source database for any replication task that you create.

> **Note** To determine if you are connected to the database you want to use or if the connection information you entered is correct, click **Test Connection**.
>
> If the connection is successful a message in green is displayed. If the connection fails, an error message is displayed at the bottom of the dialog box.
>
> To view the log entry if the connection fails, click **View Log**. The server log is displayed with the information for the connection failure. Note that this button is not available unless the test connection fails.

Adding a Non-Relational ARC Data Source to Attunity Replicate

When you add a database to Attunity Replicate and you select a non-relational ARC data source as the database type, the following dialog box opens.

To add an ARC source database to Attunity Replicate:

1. In the Attunity Replicate console, click **Manage Endpoint Connections** to open the **Add Endpoint Connections** dialog box and then click **New Endpoint Connection**. For

more information on adding an endpoint to Attunity Replicate, see Working with Endpoints.

2. In the **Name** field, type a name for your database. This can be any name that will help to identify the database being used.
3. In the **Description** field, type a description that helps to identify the ARC CDC Solution. This is optional.
4. Select **Source** as the **role**.
5. Select an ARC non-relational data source from the **Type** list. The ARC data sources are listed as <*database*> **(ARC)**, for example **RMS (ARC)**. For a list of supported non-relational data sources, see Using ARC CDC Agents as Endpoints.
6. When working with the **RMS (ARC)** data source, choose one of the following Change Processing modes:
   - **Non relational** (the default) - When this mode is selected, Replicate reads the changes from a CSV file that contains the modified data records. Use this mode if you need to retrieve changes to arrays and variant tables.

     If you select this option, continue from Step 7 below.
   - **Relational** - When this mode is selected, Replicate reads the changes directly from the ARC Agent. Relational mode improves performance but does not support changes to complex data structures such as arrays and variant tables.

     If you select this option, continue from Step 6 in Adding a Relational ARC Data Source to Attunity Replicate.
7. In the **Port** field, type the port number for the port you used when creating the CDC Router in Attunity Studio. The default port number is 2551.
8. In the **CDC Solution** field, enter the name of the solution you defined when you created the data source in Attunity Studio.
9. If a username and password are required to access the CDC Solution Router, enter them in the **User name** and **Password** fields in the **Local ARC router** section.
10. If a username and password are required to access the CDC Solution, enter them in the **User name** and **Password** fields in the **ARC on <source> machine** section.
11. Required for **IBM IMS (ARC)** only: In the **Bulk started task** field, specify the correct z/OS Started Task name for IMS/BMP or IMS/DLI. This member was copied to the z/OS PROCLIB library from <ARC HLQ>.USERLIB. NVBMPSRV and NVIMSSRV are the provided member names.

    Note If you choose IMS/DLI, you will need to close the database to IMS/TM or IMS/DBCTL. This option might be faster than using BMP. IMS/BMP does not require exclusive access to the database.

12. Click **OK** to add the database to Attunity Replicate. You can use this database as the source database for any replication task that you create.

Note To determine if you are connected to the database you want to use or if the connection information you entered is correct, click **Test Connection**.

If the connection is successful a message in green is displayed. If the connection fails, an error message is displayed at the bottom of the dialog box.

To view the log entry if the connection fails, click **View Log**. The server log is displayed with the information for the connection failure. Note that this button is not available unless the test connection fails.

Add the ARC CDC Solution Endpoint to a Task

You can use any ARC CDC Solution that you define as the source in a task. To use an ARC CDC Solution as your source, drag the ARC database from the **Endpoints** pane to your task.

For information on how to create a task, see Adding a Source and Target Endpoint to a Task.

Setting Advanced Connection Properties

In the **Advanced** tab, you can set the following properties:

- **Encryption key name**: Enter name of the encryption key defined in the **User: NAV** tab in ARC.
- **Encryption key value**: Enter value of the encryption key specified in the **Encryption key name** field above.

Note For a detailed explanation of how to encrypt session between Replicate and ARC endpoints, see Encrypting Communications Between Replicate and ARC Data Sources.

- **Fixed NAT**: Select this to indicate that the connection is made with a fixed network address translation.
- **Timeout**: Enter the amount of time, in seconds, to wait for interactions before dis-connecting. 0 indicates that the system does not timeout. The default value is **0**.
- **Event wait**: Enter the maximum amount of time (in seconds) to wait for a change event to take place before the system times out. The default value is **300**.
- **CDC batch size**: Enter the maximum number of change events that can be transferred in a single batch. The default value is **200**.
- **Bulk batch size**: Enter the unloading batch size. The default value is **100**.
- **Trace**: Select this to enable tracing for the change processing.

Internal Parameters

Internal parameters are parameters that are not exposed in the UI. You should only use them if instructed by Attunity Support.

To add internal Attunity Replicate parameters:

1. Click the **Internal Parameters** link.

   The **Internal Parameters** dialog box opens.
2. In the edit box, type the name of the parameter you need to add and then click it.
3. The parameter is added to the table below the search box with its default value.
4. Change the default value as required.
5. To reset the parameter value to its default, click the "Restore default value" icon at the end of the row.

Settings Summary

You can view a summary of your settings by clicking the **Setting Summary** link. This is useful if you need to send a summary of your settings to Attunity Support.

9 | Adding and Managing Target Endpoints

This chapter describes how to configure target endpoint settings.

In this chapter:

Using Oracle as a Target

This section describes how to set up and use an Oracle database as a target endpoint in a replication task.

In this section:

Prerequisites

Before you can work with an Oracle endpoint, make sure the prerequisites listed in this section have been met.

- On Windows systems, install Oracle Instant Client for Microsoft Windows (x64) Version 11.2.0.3.0 and above.

  Note Support for the XMLTYPE data type requires the full Oracle Client.

- On Linux systems, install Oracle Instant Client for Linux (x86-64) Version 11.2.0.3.0 and above.

  Note Support for the XMLTYPE data type requires the full Oracle Client.

  In addition, if not already included in your system, you need to create a symbolic link in the `$Oracle_Home\lib` directory. This link should be called `libclntsh.so`, and should point to a specific version of this file. For example, on an Oracle 12c client:

  ```
  lrwxrwxrwx 1 oracle oracle 63 Oct 2 14:16 libclntsh.so ->
  /u01/app/oracle/home/lib/libclntsh.so.12.1
  ```

- Additionally, the `LD_LIBRARY_PATH` environment variable should be appended with the Oracle `lib` directory and added to the `site_arep_login.sh` script.

Limitations

The following limitations apply:

- The Attunity Replicate Oracle database cannot create a new schema on the Oracle database. Therefore, if you are replicating data to an Oracle target and you want to change the schema name, the new schema name must already exist on the Oracle database. If it does not exist, you must create the schema on the database, then you can use that

schema name in Attunity Replicate.

- The **Use direct path full load** option does not support the following:
  - Tables with INDEXTYPE CONTEXT

    Workaround:

    Use Array Load.
  - Bidirectional replication
- In Batch Optimized Apply mode, loading into the net changes table uses Direct Path which does not support XMLType.

  Workaround:

  Use Transactional Apply mode.
- Attunity Replicate cannot create a new schema on the Oracle database. To replicate to a new schema, the new schema name must already exist on the Oracle target. You can then specify the new schema name in the Task Settings' Target Metadata and Control Tables tabs as required.

Security Requirements

A user must have the following privileges granted in the Oracle database to use an Oracle target in an Attunity Replicate task:

Note If any of the required privileges cannot be granted to a V$xxx, then grant them to the V_$xxx.

- SELECT ANY TRANSACTION
- SELECT on V$NLS_PARAMETERS
- SELECT on V$TIMEZONE_NAMES
- SELECT on ALL_INDEXES
- SELECT on ALL_OBJECTS
- SELECT on DBA_OBJECTS
- SELECT on ALL_TABLES
- SELECT on ALL_USERS
- SELECT on ALL_CATALOG
- SELECT on ALL_CONSTRAINTS
- SELECT on ALL_CONS_COLUMNS
- SELECT on ALL_TAB_COLS
- SELECT on ALL_IND_COLUMNS
- CREATE ANY TABLE
- DROP ANY TABLE
- SELECT ANY TABLE
- INSERT ANY TABLE

- DELETE ANY TABLE
- UPDATE ANY TABLE
- CREATE ANY VIEW
- DROP ANY VIEW
- CREATE ANY PROCEDURE
- ALTER ANY PROCEDURE
- DROP ANY PROCEDURE
- CREATE ANY SEQUENCE
- ALTER ANY SEQUENCE
- DROP ANY SEQUENCE

You can add the following permissions to use a specific table list:

- SELECT on <any-replicated-table>
- ALTER on <any-replicated-table>

The following permission must be granted for logon:

- CREATE SESSION

The following permission must be granted if you are using a direct path:

- LOCK ANY TABLE

If the "DROP and CREATE table" or "TRUNCATE before loading" option is selected in the Full Load Settings tab *and* the target table schema is different from the Attunity Replicate user, the following permission must be granted:

- DROP ANY TABLE

To store changes in Change Tables or in an Audit Table when the target table schema is different from the Attunity Replicate user, the following permission must be granted:

- CREATE ANY INDEX

The Attunity Replicate user must also be granted read permissions for the following DBA tables:

- SELECT on DBA_USERS
- SELECT on DBA_TAB_PRIVS
- SELECT on DBA_OBJECTS
- SELECT on DBA_SYNONYMS
- SELECT on DBA_SEQUENCES
- SELECT on DBA_TYPES
- SELECT on DBA_INDEXES
- SELECT on DBA_TABLES
- SELECT on DBA_TRIGGERS

Oracle Target Data Types

The Oracle database for Attunity Replicate supports most Oracle data types. The following table shows the Oracle target data types that are supported when using Attunity Replicate and the default mapping from Attunity Replicate data types.

For information on how to view the data type that is mapped from the source, see the section for the source database you are using.

For additional information about Attunity Replicate data types, see Replicate Data Types.

Table 9.1 | Supported Oracle Data Types with Mapping from Attunity Replicate **Data Types**

Attunity Replicate Data Types	Oracle Data Types
BOOLEAN	**NUMBER (1)**
BYTES	**RAW (length)**
DATE	**DATETIME**
TIME	**TIMESTAMP (0)**
DATETIME	**TIMESTAMP (scale**)
INT1	**NUMBER (3)**
INT2	**NUMBER (5)**
INT4	**NUMBER (10)**
INT8	**NUMBER (19)**
NUMERIC	**NUMBER (p,s)**
REAL4	**BINARY_FLOAT**
REAL8	**BINARY_DOUBLE**
STRING	With date indication: **DATE** With time indication: **TIMESTAMP** With timestamp indication: **TIMESTAMP** With timestamp_with_timezone indication: **TIMESTAMP WITH TIMEZONE** With timestamp_with_local_timezone indication: **TIMESTAMP WITH LOCAL TIMEZONE** With interval_year_to_month indication: **INTERVAL YEAR TO MONTH** with interval_day_to_second indication: **INTERVAL DAY TO SECOND** If Length > 4000: **CLOB**

Table 9.1 | Supported Oracle Data Types with Mapping from Attunity Replicate Data Types (Cont.)

Attunity Replicate Data Types	Oracle Data Types
	In all other cases: **VARCHAR2 (Length)**
UINT1	**NUMBER (3)**
UINT2	**NUMBER (5)**
UINT4	**NUMBER (10)**
UINT8	**NUMBER (19)**
WSTRING	**NVARCHAR2 (length)** Note that when length is greater than 2000, the column data type will be **NCLOB**.
BLOB	**BLOB** To use this data type with Attunity Replicate, you must enable the use of BLOBs for a specific task. BLOB data types are supported only in tables that include a primary key. For more information, see **LOB support** in **Task Settings**/Metadata.
CLOB	**CLOB** To use this data type with Attunity Replicate, you must enable the use of CLOBs for a specific task. During CDC, CLOB data types are supported only in tables that include a primary key. For more information, see **LOB support** in **Task Settings**/Metadata.
NCLOB	**NCLOB** To use this data type with Attunity Replicate, you must enable the use of NCLOBs for a specific task. During CDC, NCLOB data types are supported only in tables that include a primary key. For more information, see **LOB support** in **Task Settings**/Metadata.

Table 9.1 | Supported Oracle Data Types with Mapping from Attunity Replicate Data Types (Cont.)

Attunity Replicate Data Types	Oracle Data Types
The XMLTYPE target data type is only relevant in Oracle-to-Oracle replication tasks. See the note below.	**XMLTYPE**

Note When the source database is Oracle, the source data types will be replicated "as is" to the Oracle target. For example, an XMLTYPE data type on the source will be created as an XMLTYPE data type on the target.

Setting General Connection Properties

This section describes how to configure general connection properties. For an explanation of how to configure advanced connection properties, see **Setting Advanced Connection Properties** below.

Note The total number of columns per table supported in Batch optimized apply mode can be expressed using the following formula:

```
2 * columns_in_original_table + columns_in_primary_key <= 999
```

So, for example, if the original tables has 25 columns and its Primary Key consists of 5 columns, then the total number of columns would be 55. If a table exceeds the supported number of columns, Replicate will apply all of the changes in one-by-one mode.

To add an Oracle target endpoint to Attunity Replicate:

1. In the Attunity Replicate console, click **Add database** to open the Add Endpoints dialog box. For more information on adding an endpoint to Attunity Replicate, see Working with Endpoints.
2. In the **Name** field, type a name for your database. This can be any name that will help to identify the database being used.
3. In the **Description** field, type a description that helps to identify the Oracle database. This is optional.
4. Select **TARGET** as the database **role**.
5. Select **Oracle** as the database **Type**.
6. Type the Oracle **Connection String** for the Oracle database you want to work with. You can type the connect string in any Oracle format, for example:

   ```
   //host:port/service name
   ```

   Where:

- `host`: This is the name or IP address for the computer with the Oracle database that you are using. For example, `johnboy_W7` or `255.255.255.0`.
- `port`: (optional) This is the TNS Listener Port number for the computer with the Oracle database that you are using. If you do not enter a port number the default Oracle TNS Listener port is used.
- `service name`: (optional) This is the service name for the computer with the Oracle database you are using. If you do not enter a service name the default service name is used.

You can also enter an Oracle Net keyword-value pair. For example:

```
"(DESCRIPTION=(ADDRESS=(PROTOCOL=tcp) (HOST=dlsun242) (PORT=5521))
(CONNECT_DATA=(SERVICE_NAME=bjava21)))"
```

Note This information is case sensitive.

Note To determine if you are connected to the database you want to use or if the connection information you entered is correct, click **Test Connection**.

If the connection is successful a message in green is displayed. If the connection fails, an error message is displayed at the bottom of the dialog box.

To view the log entry if the connection fails, click **View Log**. The server log is displayed with the information for the connection failure. Note that this button is not available unless the test connection fails.

7. Type the Oracle authentication information (**User Name**, **Password**) for the authorized user for this Oracle database. If you do not know this information, see your Oracle database Administrator (DBA).

   To prevent illicit database activity by unauthorized third-parties, Replicate can be configured to automatically replace the user-entered password with a strong random password.

   For more information, see Configuring Replicate to Automatically Replace the User-Entered Password.

   Note This information is case sensitive.

   Important: Make sure that the Oracle user entered in the Oracle Authentication section has the correct access privileges. For information on how to provide the required privileges, see Security Requirements.

Setting Advanced Connection Properties

You can set additional properties in the **Advanced** tab of the Oracle database connection settings.

You can set the following properties:

- **Use direct path full load**: Select this to use the OCI direct path protocol for bulk loading Oracle tables. This is the default selection.

  Note Due to an issue with Oracle Direct Path, when this option is selected and **If target table already exists** is set to **Do nothing** in the Full Load Settings, the following occurs:

  - The first time the task runs, no error will be issued and rows with the same Primary Key may be added to the target table.
  - The second time the task runs, the setting will take effect.
  - Any subsequent times the task runs, an error will be generated.

Internal Parameters

Internal parameters are parameters that are not exposed in the UI. You should only use them if instructed by Attunity Support.

To add internal Attunity Replicate parameters:

1. Click the **Internal Parameters** link.

   The **Internal Parameters** dialog box opens.
2. In the edit box, type the name of the parameter you need to add and then click it.
3. The parameter is added to the table below the search box with its default value.
4. Change the default value as required.
5. To reset the parameter value to its default, click the "Restore default value" icon at the end of the row.

Settings Summary

You can view a summary of your settings by clicking the **Setting Summary** link. This is useful if you need to send a summary of your settings to Attunity Support.

Using Microsoft SQL Server as a Target

This section describes how to set up and use a Microsoft SQL Server database as a target in a replication task.

In this section:

Supported Editions

Attunity Replicate supports the following Microsoft SQL Server editions:

- Enterprise Edition
- Standard Edition
- Workgroup Edition
- Developer Edition

Prerequisites

Make sure the following prerequisites have been met:

- Client prerequisites (for source and target endpoints):

  Attunity Replicate for Windows:

  For all versions of Microsoft SQL Server, Microsoft SQL Server Native Client 11.0 must be installed on the Attunity Replicate Server machine.

  Attunity Replicate for Linux:

  First, install Microsoft ODBC Driver 13.1 for Linux on the Attunity Replicate Server machine.

  Then, on the Attunity Replicate Server machine, open a Unix shell and perform the following steps:

  1. Change the working directory to:
     ```
     cd <product_dir>/bin
     ```
  2. Stop the Replicate service:
     ```
     /arep.ctl stop
     ```
  3. Optionally, confirm that the service has stopped:

```
ps -ef | grep repctl
```

4. Create the `site_arep_login.sh` file:
   ```
   touch site_arep_login.sh
   ```
5. Copy the driver location to the `site_arep_login.sh` file:
   ```
   echo "export LD_LIBRARY_PATH=$LD_LIBRARY_
   PATH:/opt/microsoft/msodbcsql/lib64/" > site_arep_login.sh
   ```
6. Optionally, confirm that the driver location was copied:
   ```
   cat site_arep_login.sh
   ```
7. Start the Replicate service:
   ```
   ./arep.ctl start
   ```
8. Optionally confirm that the service has started:
   ```
   ps -ef | grep repctl
   ```

Note Replicate requires the following ODBC library:

`libmsodbcsql-13.1.so.0.0`

If the existing library has a different version number (e.g. `libmsodbcsql-13.1.so.0.2`), you need to create a symbolic link between the existing library and the required library.

To check which library version is currently installed

Issue the following command:

```
cd /opt/microsoft/msodbcsql/lib64/
```

To create a symbolic link

Issue the following command:

```
ln -s <existing_library_name> libmsodbcsql-13.1.so.0.0
```

where `<existing_library_name>` is the name of the currently installed library (e.g. `libmsodbcsql-13.1.so.0.2`).

- A Microsoft SQL Server account with the specific access privileges is required. See Source Permissions or Target Permissions for more information.
- Microsoft SQL Server as a source must be configured for a full backup to work with Attunity Replicate. For more information, see Preparing Microsoft SQL Server Backup and Recovery.

Limitations

When using a Microsoft SQL Server database as a target in a Replicate task, The following imitations apply:

- When **Use BCP for loading tables** is selected in the **Advanced** tab (the default), unlimited LOB columns are not supported in **Batch optimized apply** change processing mode. You can work around this limitation by limiting LOB column size in the task settings, clearing the **Use BCP for loading tables** option or switching to **Transactional apply** mode.
- When the **Use BCP for loading tables** option is enabled in the **Advanced** tab, triggers are not executed.
- When Attunity Replicate is installed on Linux, the **Use BCP for loading tables** option in the **Advanced** tab is not supported.
- Microsoft SQL Server 2012 Target: When the target table is created manually with a computed column, Full Load replication is not supported in BCP mode. Disabling the "Use BCP for loading tables" option in the **Advanced** tab will resolve this issue. For more information on BCP mode, see Setting Advanced Connection Properties.

Permissions

The following describes the security requirements for using Attunity Replicate with a Microsoft SQL Server target.

The Attunity Replicate user must have at least the `db_owner` user role on the Microsoft SQL Server database you are connecting to.

A Microsoft SQL Server system administrator must provide this permission for all Attunity Replicate users.

Microsoft SQL Server Target Data Types

The Microsoft SQL Server target for Attunity Replicate supports most Microsoft SQL Server data types. The following table shows the Microsoft SQL Server target data types that are supported when using Attunity Replicate and the default mapping from Attunity Replicate data types.

For information on how to view the data type that is mapped from the source, see the section for the source database you are using.

For additional information about Attunity Replicate data types, see Replicate Data Types.

Table 9.2 | Microsoft SQL Server Target Data Types with Mapping from Attunity Replicate **Data Types**

Attunity Replicate Data Types	Microsoft SQL Server Data Types
BOOLEAN	**TINYINT**

Table 9.2 | Microsoft SQL Server Target Data Types with Mapping from Attunity Replicate Data Types (Cont.)

Attunity Replicate Data Types	Microsoft SQL Server Data Types
BYTES	**VARBINARY(length)**
DATE	For Microsoft SQL Server 2008 and later: **DATE** For earlier versions: If scale is < or = 3: **DATETIME** In all other cases: **VARCHAR (37)**
TIME	For Microsoft SQL Server 2008 and later: **DATETIME2 (%d)** For earlier versions: If scale is < or = 3: **DATETIME** In all other cases: **VARCHAR (37)**
DATETIME	For Microsoft SQL Server 2008 and later: **DATETIME2 (scale)** For earlier versions: If scale is < or = 3: **DATETIME** In all other cases: **VARCHAR (37)**
INT1	**SMALLINT**
INT2	**SMALLINT**
INT4	**INT**
INT8	**BIGINT**
NUMERIC	**NUMERIC (p,s)**
REAL4	**REAL**
REAL8	**FLOAT**
STRING	If column is date or time then: For Microsoft SQL Server 2008 and later: **DATETIME2** For earlier versions: If scale is < or = 3: **DATETIME** In all other cases: **VARCHAR (37)** If the column is *not* a date or time: **VARCHAR (length)**

Table 9.2 | Microsoft SQL Server Target Data Types with Mapping from Attunity Replicate Data Types (Cont.)

Attunity Replicate Data Types	Microsoft SQL Server Data Types
UINT1	**TINYINT**
UINT2	**SMALLINT**
UINT4	**INT**
UINT8	**BIGINT**
WSTRING	**NVARCHAR (length)**
BLOB	**VARBINARY (max)** **IMAGE** To use this data type with Attunity Replicate, you must enable the use of BLOBs for a specific task. BLOB data types are supported only in tables that include a primary key. For more information, see **LOB support** in **Task Settings**/Metadata.
CLOB	**VARCHAR (max)** **TEXT** To use this data type with Attunity Replicate, you must enable the use of CLOBs for a specific task. During CDC, CLOB data types are supported only in tables that include a primary key. For more information, see **LOB support** in **Task Settings**/Metadata.
NCLOB	**NVARCHAR (max)** **NTEXT** To use this data type with Attunity Replicate, you must enable the use of NCLOBs for a specific task. During CDC, NCLOB data types are supported only in tables that include a primary key. For more information, see **LOB support** in **Task Settings**/Metadata.

Setting General Connection Properties

This section describes how to configure general connection properties. For an explanation of how to configure advanced connection properties, see **Setting Advanced Connection Properties** below.

To add a Microsoft SQL Server target endpoint to Attunity Replicate:

1. In the Attunity Replicate Console, click **Manage Endpoint Connections** to open the **Manage Endpoints Connections** dialog box and then click **New Endpoint Connection**. For more information on adding an endpoint to Attunity Replicate, see Working with Endpoints.
2. In the **Name** field, type a name for your database. This can be any name that will help to identify the database being used.
3. In the **Description** field, type a description that helps to identify the Microsoft SQL Server database. This is optional.
4. Select **TARGET** as the database **role**.
5. Select **Microsoft SQL Server** as the database **Type**.
6. Specify the **Server name**. This is the host name or IP address of the computer with the Microsoft SQL Server instance containing the target database.

   Note To override the default port, add the port to the server name, separated by a comma. For example, if the server name is **`myserver.company.local`** and the port is **`3333`**, then the server name should be written like this:

   ```
   myserver.company.local,3333
   ```

7. Select **Windows authentication** (only relevant when Replicate is installed on Windows) or **SQL Server authentication**.

   If you select **Windows authentication**, you will work with the user credentials for the Windows domain. This privilege must be configured in the Microsoft SQL Server database by the system administrator. Note that this option is not relevant when Microsoft SQL Server is running on Linux.

   Note When using Windows authentication, make sure that the user account that is associated with the Attunity Replicate **Server** service has Network read and write permissions. This must be configured by a Windows system administrator.

   See also Working with Windows Authentication.

   If you select **SQL Server authentication**, type the Microsoft SQL Server authentication information (User name, Password) for the authorized user for this Microsoft SQL Server database. If you do not know this information, see the Microsoft SQL Server System Administrator.

   To prevent illicit database activity by unauthorized third-parties, Replicate can be configured to automatically replace the user-entered password with a strong random password.

   For more information, see Configuring Replicate to Automatically Replace the User-Entered Password.

Note This information is case sensitive.

Important: Make sure that the Microsoft SQL Server user has the correct access privileges. For information on how to provide the required privileges, see Required Permissions.

Note To determine if you are connected to the database you want to use or if the connection information you entered is correct, click **Test Connection**.

If the connection is successful a message in green is displayed. If the connection fails, an error message is displayed at the bottom of the dialog box.

To view the log entry if the connection fails, click **View Log**. The server log is displayed with the information for the connection failure. Note that this button is not available unless the test connection fails.

8. Type the **Database name** or click **Browse** and select one from the list of available databases. This is the name of the database from where you are replicating the data.

Setting Advanced Connection Properties

In the **Advanced** tab, you can set the following properties:

- **Use BCP for loading tables**: Select this to transfer data for full-load operations using BCP.

  Note
  - When the target table contains an identity column that does not exist in the source table, you must disable the **Use BCP for loading tables** option.
  - BCP is not supported by when Replicate is installed on Linux.

- **BCP packet size**: The maximum size of the packets (in bytes) used to transfer data using BCP.
- **Filegroup for Attunity Replicate internal tables:** Optionally, specify a filegroup for the Attunity Replicate internal tables. When the replication task starts, all of the internal Attunity Replicate control tables will be created in the specified filegroup.

  The following is an example of a command for creating a filegroup:

```
ALTER database replicate
ADD FILEGROUP Test1FG1;
GO
ALTER database replicate
ADD FILE
(
```

```
NAME = test1dat5,
 FILENAME = 'C:\temp\DATA\t1dat5.ndf',
 SIZE = 5MB,
 MAXSIZE = 100MB,
 FILEGROWTH = 5MB
)
TO FILEGROUP Test1FG1;
GO
```

- **Additional ODBC connection properties**: Specify any additional ODBC connection parameters that you want to use.

Internal Parameters

Internal parameters are parameters that are not exposed in the UI. You should only use them if instructed by Attunity Support.

To add internal Attunity Replicate parameters:

1. Click the **Internal Parameters** link.
   The **Internal Parameters** dialog box opens.
2. In the edit box, type the name of the parameter you need to add and then click it.
3. The parameter is added to the table below the search box with its default value.
4. Change the default value as required.
5. To reset the parameter value to its default, click the "Restore default value" icon at the end of the row.

Settings Summary

You can view a summary of your settings by clicking the **Setting Summary** link. This is useful if you need to send a summary of your settings to Attunity Support.

Using SAP Sybase ASE as a Target

This section describes how to set up and use a SAP Sybase ASE database as the target endpoint in a replication task.

In this section:

Prerequisites

Make sure the following prerequisites have been met:

- SAP Sybase ASE ODBC 64-bit client installed on the computer where Attunity Replicate is located.
- SAP Sybase ASE replication enabled for tables using the `sp_setreptable` command or privileges to enable it automatically.
- `RepAgent` must be disabled on the SAP Sybase ASE database.
- When replicating to SAP Sybase ASE 15.7 installed on a Windows machine configured with a non-Latin language (e.g. Chinese), Attunity Replicate requires Sybase 15.7 SP121 to be installed on the SAP Sybase ASE machine.

Limitations

The following limitations apply:

- Only one Attunity Replicate task can be run per SAP Sybase ASE database.
- Attunity Replicate tasks cannot run concurrently with SAP Sybase ASE Replication Server against the same SAP Sybase ASE database.
- Zero values located at the end of binary data type strings are truncated when replicated to the target database. For example, `0x00000000000000000000000001000001000000000` in the source table will become `0x00000000000000000000000001000001` in the target table.
- Attunity Replicate creates the target table with columns that do not allow NULL values, if the database default is *not* to allow NULL values. Consequently, if a Full Load or CDC replication task contains empty values, errors will occur.

  To prevent this from happening:

- Right-click the database name and select **Properties** from the context menu.
- In the **Options** tab, select **Allow nulls by default** and then click **OK**.

Security Requirements

You must provide SAP Sybase ASE account access to the Attunity Replicate user. This user must have read/write privileges in the SAP Sybase ASE database.

SAP Sybase ASE Database Target Data Types

The following table shows the SAP Sybase ASE database target data types that are supported when using Attunity Replicate and the default mapping from Attunity Replicate data types.

Note SAP Sybase ASE does not support applying changes to binary data types in **Batch optimized apply** mode. For more information on **Batch optimized apply** mode, see Changes Processing Tuning.

For information on how to view the data type that is mapped in the target, see the section for the target database you are using.

For additional information about Attunity Replicate data types, see Replicate Data Types.

Table 9.3 | Supported SAP Sybase ASE Data Types with Mapping from Attunity Replicate **Data Types**

Attunity Replicate Data Types	SAP Sybase ASE Data Types
BOOLEAN	BIT
BYTES	VARBINARY (Length)
DATE	DATE
TIME	TIME
DATETIME	If scale is => 0 and =< 6, then: BIGDATETIME If scale is => 7 and =< 9, then: VARCHAR (37)
INT1	TINYINT
INT2	SMALLINT
INT4	INTEGER
INT8	BIGINT
NUMERIC	NUMERIC (p,s)

Table 9.3 | Supported SAP Sybase ASE Data Types with Mapping from Attunity Replicate Data Types (Cont.)

Attunity Replicate Data Types	SAP Sybase ASE Data Types
REAL4	REAL
REAL8	DOUBLE PRECISION
STRING	VARCHAR (Length)
UINT1	TINYINT
UINT2	UNSIGNED SMALLINT
UINT4	UNSIGNED INTEGER
UINT8	UNSIGNED BIGINT
WSTRING	VARCHAR (Length)
BLOB	IMAGE
CLOB	UNITEXT
NCLOB	TEXT

Non-Supported Data Types

Target SAP Sybase ASE tables with columns of the following SAP Sybase ASE data types cannot be replicated. Replicated columns with these data types will show as null.

- UDT

Setting General Connection Properties

This section describes how to configure general connection properties. For an explanation of how to configure advanced connection properties, see **Setting Advanced Connection Properties** below.

To add a SAP Sybase ASE target endpoint to Attunity Replicate:

1. In the Attunity Replicate Console, click **Manage Endpoint Connections** to open the **Manage Endpoints Connections** dialog box. Then click the **New Endpoint Connection** button. For more information on adding an endpoint to Attunity Replicate, see Working with Endpoints.
2. In the **Name** field, type a name for your database. This can be any name that will help to identify the database being used.
3. In the **Description** field, type a description that helps to identify the SAP Sybase ASE database. This is optional.
4. Select **TARGET** as the database **role**.
5. Select **SAP Sybase ASE** as the database **Type**.
6. In the **Server Name** field, enter the host name or IP address of the computer on which

the SAP Sybase ASE database is installed.

Note Consider the following:

- This information is case sensitive.
- You can use the **Advanced** tab to add specific properties and create a custom connect string. In this case, you do not need to enter information in this tab. For more information on using the **Advanced** tab, see Setting Advanced Connection Properties.
- To determine if you are connected to the database you want to use or if the connection information you entered is correct, click **Test Connection**.

  If the connection is successful a message in green is displayed. If the connection fails, an error message is displayed at the bottom of the dialog box.

  To view the log entry if the connection fails, click **View Log**. The server log is displayed with the information for the connection failure. Note that this button is not available unless the test connection fails.

7. Optionally, change the default port (5000).
8. Type the SAP Sybase ASE authentication information (**User Name**, **Password**) for the authorized user for this SAP Sybase ASE database. If you do not know this information, see your SAP Sybase ASE database Administrator (DBA).

Note Consider the following:

- This information is case sensitive.
- This information is required. If you are using the **Advanced** tab to create a custom string, make sure to include the **User Name** and **Password** properties. See Setting Advanced Connection Properties for more information.
- If you want to set custom properties for this database, see Setting Advanced Connection Properties.

Important: Make sure that the SAP Sybase ASE user entered in the SAP Sybase ASE Authentication section has the correct access privileges. For information on how to provide the required privileges, see Security Requirements.

9. In the **Database name** field, enter the SAP Sybase ASE database name.

Setting Advanced Connection Properties

In the **Advanced** tab, you can set the following properties:

- **Additional ODBC connection properties**: Specify any additional ODBC connection parameters that you want to use.

Note If the user name or password specified in the **General** tab contains non-Latin characters (e.g. Chinese), the following property is required:

```
charset=gb18030
```

Internal Parameters

Internal parameters are parameters that are not exposed in the UI. You should only use them if instructed by Attunity Support.

To add internal Attunity Replicate parameters:

1. Click the **Internal Parameters** link.

   The **Internal Parameters** dialog box opens.
2. In the edit box, type the name of the parameter you need to add and then click it.
3. The parameter is added to the table below the search box with its default value.
4. Change the default value as required.
5. To reset the parameter value to its default, click the "Restore default value" icon at the end of the row.

Settings Summary

You can view a summary of your settings by clicking the **Setting Summary** link. This is useful if you need to send a summary of your settings to Attunity Support.

Using MySQL as a Target

This section describes how to set up and use a MySQL database as the target endpoint in a replication task.

Note Attunity Replicate also supports MariaDB as a target endpoint. The procedures for configuring connectivity to MariaDB are identical to those described for MySQL.

In this section:

Security Requirements

You must provide MySQL account access to the Attunity Replicate user. This user must have read/write privileges in the MySQL database.

MySQL Database Target Data Types

The following table shows the MySQL database target data types that are supported when using Attunity Replicate and the default mapping from Attunity Replicate data types.

For information on how to view the data type that is mapped in the target, see the section for the target database you are using.

For additional information about Attunity Replicate data types, see Replicate Data Types.

Table 9.4 | Supported MySQL Data Types with Mapping from Attunity Replicate Data Types

Attunity Replicate Data Types	MySQL Data Types
BOOL	BOOL
BYTES	If length is => 1 and =< 8095, then: VARBINARY (Length) If length is => 8096 and =< 65535, then: BLOB If length is => 65536 and =< 16777215, then: MEDIUMBLOB If length is => 16777216 and =< 2147483647, then:

Table 9.4 | Supported MySQL Data Types with Mapping from Attunity Replicate Data Types (Cont.)

Attunity Replicate Data Types	MySQL Data Types
	LONGLOB
DATE	DATE
TIME	TIME
DATETIME	If scale is => 0 and =< 6, then: DATETIME (Scale) If scale is => 7 and =< 9, then: VARCHAR (37)
INT1	TINYINT
INT2	SMALLINT
INT4	INTEGER
INT8	BIGINT
NUMERIC	If scale is => 0 and =< 30, then: DATETIME (p,s) If scale is => 31 and =< 100, then: VARCHAR (45)
REAL4	FLOAT
REAL8	DOUBLE
STRING	If length is => 1 and =< 8095, then: VARCHAR (Length) If length is => 8096 and =< 65535, then: TEXT If length is => 65536 and =< 16777215, then: MEDIUMTEXT If length is => 16777216 and =< 2147483647, then: LONGTEXT
UINT1	UNSIGNED TINYINT
UINT2	UNSIGNED SMALLINT
UINT4	UNSIGNED INTEGER
UINT8	UNSIGNED BIGINT
WSTRING	If length is => 1 and =< 8095, then: VARCHAR (Length)

Table 9.4 | Supported MySQL Data Types with Mapping from Attunity Replicate Data Types (Cont.)

Attunity Replicate Data Types	MySQL Data Types
	If length is => 8096 and =< 65535, then: TEXT If length is => 65536 and =< 16777215, then: MEDIUMTEXT If length is => 16777216 and =< 2147483647, then: LONGTEXT
BLOB	If length is => 1 and =< 65535, then: BLOB If length is => 65536 and =< 2147483647, then: LONGBLOB If length is => 0 and =< 0, then: LONGBLOB (Full Lob Support)
NCLOB	If length is => 1 and =< 65535, then: TEXT If length is => 65536 and =< 2147483647, then: LONGTEXT - CHARACTER SET: ucs2 If length is => 0 and =< 0, then: LONGTEXT - CHARACTER SET: ucs2 (Full Lob Support)
CLOB	If length is => 1 and =< 65535, then: TEXT If length is => 65536 and =< 2147483647, then: LONGTEXT If length is => 0 and =< 0, then: LONGTEXT (Full Lob Support)

Setting up a MySQL Database as a Target in Attunity Replicate

This section describes how to configure general connection properties. For an explanation of how to configure advanced connection properties, see **Setting Advanced Connection Properties** below.

To add a MySQL target endpoint to Attunity Replicate:

1. In the Attunity Replicate Console, click **Manage Endpoint Connections** to open the **Manage Endpoints Connections** dialog box. Then click the **New Endpoint**

Connection button. For more information on adding an endpoint to Attunity Replicate, see Working with Endpoints.

2. In the **Name** field, type a name for your database. This can be any name that will help to identify the database being used.
3. In the **Description** field, type a description that helps to identify the MySQL database. This is optional.
4. Select **TARGET** as the database **role**.
5. Select **MySQL** as the database **Type**.
6. In the **Server** field, enter the host name or IP address of the computer on which the MySQL database is installed.

   Note Consider the following:
   - This information is case sensitive.
   - You can use the **Advanced** tab to add specific properties and create a custom connect string. In this case, you do not need to enter information in this tab. For more information on using the **Advanced** tab, see Setting Advanced Connection Properties.
   - To determine if you are connected to the database you want to use or if the connection information you entered is correct, click **Test Connection**.

     If the connection is successful a message in green is displayed. If the connection fails, an error message is displayed at the bottom of the dialog box.

     To view the log entry if the connection fails, click **View Log**. The server log is displayed with the information for the connection failure. Note that this button is not available unless the test connection fails.

7. Optionally, change the default port (3306).
8. Type the MySQL authentication information (**User Name**, **Password**) for the authorized user for this MySQL database. If you do not know this information, see your MySQL database Administrator (DBA).

   Note Consider the following:
   - This information is required. If you are using the **Advanced** tab to create a custom string, make sure to include the **User Name** and **Password** properties. See Setting Advanced Connection Properties for more information.
   - This information is case sensitive.
   - If you want to set custom properties for this database, see Setting Advanced Connection Properties.

   Important: : Make sure that the MySQL user entered in the MySQL Authentication section has the correct access privileges. For information on how to provide the

required privileges, see Security Requirements.

9. Select one of the following **Load source schemas into** options:
   - **The following database** - When this option is selected, all source schemas will be loaded into the selected database.
   - **Multiple endpoints** - When this option is selected, each of the source schemas will be loaded into its corresponding database.

Setting Advanced Connection Properties

In the **Advanced** tab, you can set the following properties:

- **Max file size (KB)**: Select or type the maximum size (in KB) of a CSV file before it is loaded into the MySQL target database. The default value is 32000 KB.
- **Use parallel loading**: Select this option to improve performance when loading data into the MySQL target database.
  - **Use the following number of threads**: Specify how many threads to use to load the data into the MySQL target database. Note that setting a large number of threads may have an adverse effect on database performance since a separate connection is required for each thread.

Note Attunity Replicate assumes that MySQL Client 5.2.6 to 5.3.x for Linux or MySQL ODBC Client 5.2.6 to 5.3.x 64-bit for Windows is installed on the Attunity Replicate Server machine. If a version later than 5.3.x is installed, you need to specify the version number as an internal parameter where `driver` is the **Parameter** and `MySQL ODBC <version> Unicode Driver` is the **Value** (where **<version>** is the client version e.g. 5.4).

For instructions on setting internal parameters, see Internal Parameters below.

Internal Parameters

Internal parameters are parameters that are not exposed in the UI. You should only use them if instructed by Attunity Support.

To add internal Attunity Replicate parameters:

1. Click the **Internal Parameters** link.

   The **Internal Parameters** dialog box opens.
2. In the edit box, type the name of the parameter you need to add and then click it.
3. The parameter is added to the table below the search box with its default value.
4. Change the default value as required.
5. To reset the parameter value to its default, click the "Restore default value" icon at the end of the row.

Settings Summary

You can view a summary of your settings by clicking the **Setting Summary** link. This is useful if you need to send a summary of your settings to Attunity Support.

Using Hadoop as a Target

This section describes how to set up and use Hadoop as the target endpoint in a replication task.

In this section:

Prerequisites

Before you begin to work with a Hadoop cluster as a target in Attunity Replicate, make sure that the following prerequisites have been met:

- **General:**
  - The Hadoop WebHDFS must be accessible from the Attunity Replicate machine.
  - The Hadoop Data Nodes must be accessible from the Attunity Replicate machine.
  - The Hadoop WebHDFS service must be running.
  - To access Hive using WebHCat, the Hadoop WebHCat service must be running. Other methods for accessing Hive are described later in this chapter.
  - The user specified in the Attunity Replicate Hadoop target settings must have access to HCatalog.
- **ODBC Access:**

  When accessing Hive using ODBC, the following ODBC drivers are supported:
  - **Hortonworks:** ODBC driver 2.1.2 and above
  - **Cloudera:** ODBC driver 2.5.19 and above

    Note Cloudera ODBC drivers 2.5.20 and above do not support the Snappy compression method.
  - **MapR:** ODBC driver 2.1.8 and above
  - **Amazon EMR:** Amazon Hive ODBC driver 1.1.1.1001
- **SSL:** Before you can use SSL, you first need to perform the following tasks:

- Configure each NameNode and each DataNode with an SSL certificate (issued by the same CA).
- Place the CA certificate on the Replicate Server machine. The certificate should be a base64-encoded PEM (OpenSSL) file.

- **Permissions:** The user specified in the Hadoop target settings must have write permission for the specified HDFS target directory.

Prerequisites for using the Cloudera Distribution as a Hadoop Target

If you are replicating to a Cloudera Hadoop Distribution and you want to use **Snappy** compression and/or set a **File Format** that is **not** Text, you first need to install Cloudera's Hive ODBC driver on the Replicate Server machine. Then configure the Hadoop target endpoint to access Hive using ODBC. For more information on this setting, see Setting General Connection Properties

See also Prerequisites for using a Linux ODBC Driver.

Prerequisites for using Amazon EMR as a Hadoop Target

- Install the Amazon Hive ODBC driver on the Replicate Server machine.
- Configure the Hadoop target endpoint to access Hive using ODBC. For more information on this setting, see Setting General Connection Properties
- Add the Names and Public IP addresses of the EMR cluster nodes to the hosts file on the Replicate Server machine. This is necessary because the names of the EMR cluster nodes are internal to AWS.

Prerequisites for using a Linux ODBC Driver

To use a Linux ODBC driver, make sure to:

- Install the latest 64-bit ODBC driver for your Hadoop distribution on the Replicate Server machine.
- *After* the driver is installed: Edit the <distribution>.hiveodbc.ini file as follows:
  - `DriverManagerEncoding=UTF-16`
  - `ODBCInstLib=libodbcinst.so`

See also Setting General Connection Properties.

Limitations

The following limitations apply:

- UPDATE/DELETE DMLs are not supported during change processing. If an UPDATE/DELETE DML was captured on the source, it will be ignored on the target and a warning will be written to the log. If the **Store Changes** option is enabled in the task settings, these records will be written to the Change Table.
- Limited LOB support only.
- Dropping columns and changing column data types are not supported.

- Due to a Hive limitation, in Hive version 0.12 and earlier versions, only alphanumeric and underscore characters are allowed in table and column names. From Hive 0.13, column names can contain any Unicode character.
- When loading data into partitioned tables, the following limitations apply:
  - The Apply Changes replication option is not supported.
  - Data can only be loaded into tables with existing partitions.
  - The **Drop and Create table** option in the task settings' Full Load Settings tab should not be selected.

    See also Support for Partitions, Buckets and Skews.
- During Change Processing, changes to source column names are not propagated to the target.
- The following Control Tables are not supported as they require UPDATE/DELETE operations (which are not supported by the Hadoop target endpoint):
  - **Replication Status** (requires UPDATE).

    Name on target: `attrep_status`
  - **Suspended Tables** (requires DELETE).

    Name on target: `attrep_suspended_tables`

    For more information on Control Tables, see Control Tables.
- Primary keys are not supported on Hive versions prior to 2.1. From Hive 2.1, primary keys are supported when accessing Hive using ODBC only.

Change Data Partitioning on Hadoop

When Change Data Partitioning is enabled, the Replicate Change Tables in Hive are partitioned by the `partition_name` column. Data files are uploaded to HDFS, according to the maximum size and time definition, and then stored in a directory under the Change Table directory. Whenever the specified partition timeframe ends, a partition is created in Hive, pointing to the HDFS directory.

Information about the partitions is written to the attrep_cdc_partitions Control Table.

Prerequisites

The prerequisites for using Change Data Partitioning with a Hadoop target endpoint are as follows:

- The target file format must be set to Text or Sequence
- Hive access must be set to ODBC

Security Requirements

The Hadoop NameNode (and data nodes when using WebHDFS) must be accessible from the Attunity Replicate machine and the user specified in the Hadoop target settings must have write permission for the specified HDFS target directory.

Hadoop Endpoint Target Data Types

The following table shows the Hadoop endpoint target data types that are supported when using Attunity Replicate and the default mapping from Attunity Replicate data types.

For information on how to view the data type that is mapped in the target, see the section for the target endpoint you are using.

For additional information about Attunity Replicate data types, see Replicate Data Types.

Table 9.5 | Supported Hadoop Data Types with Mapping from Attunity Replicate Data Types

Attunity Replicate Data Types	Hadoop Data Types
BOOL	BOOLEAN
BYTES	STRING
TIME	TIMESTAMP
DATETIME	TIMESTAMP
DATE	DATE

Note When Avro is selected as the Target storage format, the TIMESTAMP and DATE data types (which are not supported by Avro) are mapped to VARCHAR(37).

INT1	TINYINT
INT2	SMALLINT
INT4	INT
INT8	BIGINT
NUMERIC	Hive 0.13 and above: DECIMAL (p,s) Hive 0.12 and below: DECIMAL
REAL4	FLOAT
REAL8	DOUBLE
STRING	Hive 0.13 and above: VARCHAR (Length) Hive 0.12 and below: STRING
UINT1	SMALLINT
UINT2	INT

Table 9.5 | Supported Hadoop Data Types with Mapping from Attunity Replicate Data Types (Cont.)

Attunity Replicate Data Types	Hadoop Data Types
UINT4	BIGINT
UINT8	Hive 0.13 and above: DECIMAL (20,0) Hive 0.12 and below: DECIMAL
WSTRING	Hive 0.13 and above: VARCHAR (Length) Hive 0.12 and below: STRING
BLOB	STRING
NCLOB	STRING
CLOB	STRING

Setting General Connection Properties

This section describes how to configure general connection properties. For an explanation of how to configure advanced connection properties, see **Setting Advanced Connection Properties** below.

Support for Partitions, Buckets and Skews To load data into tables with partitions, buckets or skews, you first need to perform the procedure described below.

To load data into tables with partitions, buckets or skews:

1. Create the tables in Hive with these attributes (partitions, buckets or skews) *prior* to running the task.
2. Add the following values to the `hive.security.authorization.sqlstd.confwhitelist.append` property in the Hive configuration file:

   If the target tables are partitioned:

   `|hive.exec.dynamic.partition|hive.exec.dynamic.partition.mode`

   If the target tables have buckets:

   `|hive.enforce.bucketing`

   If the target tables have skews:

   `|hive.mapred.supports.subdirectories`

Note If the value(s) already exist in the `hive.security.authorization.sqlstd.confwhitelist property`, you do not need to add them to the `hive.security.authorization.sqlstd.confwhitelist.append` property.

3. Set the **Target Table Preparation** task setting to **Truncate before loading** or **Do nothing**. For more information on this setting, see Full Load Settings.

To add a Hadoop target endpoint to Attunity Replicate:

1. In the Attunity Replicate console, click **Manage Endpoint Connections** to open the **Manage Endpoint Connections** dialog box.

   For more information on adding an endpoint to Attunity Replicate, see Working with Endpoints.
2. In the **Name** field, type a name for your endpoint. This can be any name that will help to identify the endpoint being used.
3. In the **Description** field, type a description that helps to identify the Hadoop endpoint. This is optional.
4. Select **Hadoop** as the endpoint **Type**.
5. In the **Hadoop NameNode** field, enter the host name or IP address of the Hadoop NameNode machine.

   Note Consider the following:
   - This information is case sensitive.
   - To determine if you are connected to the endpoint you want to use or if the connection information you entered is correct, click **Test Connection**.

     If the connection is successful, a confirmation message is displayed. If the connection fails, an error is displayed at the bottom of the dialog box.

     To view the log entry if the connection fails, click **View Log**. The server log is displayed with the information for the connection failure. Note that this button is not available unless the test connection fails.
6. In the **Security** section, do the following:
   a. To encrypt the data between the Replicate machine and HDFS, select **Use SSL**. In order to use SSL, first make sure that the SSL prerequisites described in Prerequisites been met.

      In the **CA path** field, either specify the directory containing the CA certificate.

      -OR-

      Specify the full path to a specific CA certificate.
   b. Select one of the following authentication types:

- **User name** - Select to connect to the Hadoop cluster with only a user name. Then, in the **User name** field, specify the name of a user authorized to access the Hadoop cluster.
- **Kerberos** - Select to authenticate against the Hadoop cluster using Kerberos. Replicate automatically detects whether Attunity Replicate Server is running on Linux or on Windows and displays the appropriate settings.

  Attunity Replicate Server on Linux:

  When Attunity Replicate Server is running on Linux, select either **Ticket** or Keytab from the **Kerberos options** drop-down list.

  If you selected **Ticket**, select one of the following options:

  - **Use global Kerberos ticket file** - Select this option if you want to use the same ticket for several Hadoop endpoints (source or target). In this case, you must make sure to select this option for each Hadoop endpoint instance that you define.
  - **Use specific Kerberos ticket file** - Select this option if you want to use a different ticket file for each Hadoop endpoint (source or target). Then specify the ticket file name in the designated field.

    This option is especially useful if you need to perform a task-level audit of Replicate activity (using a third-party tool) on the Hadoop NameNode. To set this up, define several instances of the same Hadoop endpoint and specify a unique Kerberos ticket file for each instance. Then, for each task, simply select a different Hadoop endpoint instance.

  Note You need to define a global Kerberos ticket file even if you select the **Use specific Kerberos ticket file** option. The global Kerberos ticket file is used for authentication when selecting a Hive endpoint, when testing the connection (using the **Test Connection** button), and when selecting which tables to replicate.

  Note When replicating from a Hadoop source endpoint to a Hadoop target endpoint, both endpoints must be configured to use the same ticket file.

  For additional steps required to complete setup for Kerberos ticket-based authentication, see Using Kerberos Authentication.

  If you selected **Keytab**, provide the following information:

  - **Realm:** The name of the realm in which your Hadoop cluster resides.

    For example, if the full principal name is `john.doe@EXAMPLE.COM`, then `EXAMPLE.COM` is the realm.
  - **Principal:** The user name to use for authentication. The principal must be a member of the realm entered above.

For example, if the full principal name is `john.doe@EXAMPLE.COM`, then `john.doe` is the principal.

- **Keytab file**: The full path of the Keytab file. The Keytab file should contain the key of the **Principal** specified above.

Attunity Replicate Server on Windows:

When Attunity Replicate Server is running on Windows, select one of the following:

- **Use the following KDC:** Select **Active Directory** (default) if your KDC is Microsoft Active Directory or select **MIT** if your KDC is MIT KDC running on Linux/UNIX.

  Note When the Replicate KDC and the Hadoop KDC are in different domains, a relationship of trust must exist between the two domains.

- **Realm:** The name of the realm/domain in which your Hadoop cluster resides (where *realm* is the MIT term while *domain* is the Active Directory term).
- **Principal:** The username to use for authentication. The principal must be a member of the realm/domain entered above.
- When **Active Directory** is selected - **Password:** The password for the principal entered above.
- When **MIT** is selected - **Keytab file**: The keytab file containing the principal entered above.

Note When replicating from a Hadoop source endpoint to a Hadoop target endpoint, both endpoints must be configured to use the same parameters (KDC, realm, principal, and password).

If you are unsure about any of the above, consult your IT/security administrator.

For additional steps required to complete setup for Kerberos authentication, see Using Kerberos Authentication.

- **User name and password** - Select to connect to the Hadoop NameNode or to the Knox Gateway (when enabled - see below) with a user name and password. Then, in the **User name** and **Password** fields, specify the required user name and password.

  Note Consider the following:
  - A user name and password is required to access the MapR Control System.
  - This information is case sensitive.

  Important: Make sure that the specified user has the required Hadoop access privileges. For information on how to provide the required privileges, see

Security Requirements.

7. If you need to access the Hortonworks Hadoop distribution through a Knox Gateway, select **Use Knox Gateway**. Then provide values for the following fields:

   Note To be able to select this option, first select **Use SSL** and then select **Password** from the **Authentication type** drop-down list.

   - **Knox Gateway host** - The FQDN (Fully Qualified Domain Name) of the Knox Gateway host.
   - **Knox port** - The port number to use to access the host. The default is "8443".
   - **Knox Gateway path** - The context path for the gateway. The default is "gateway".

     Note The port and path values are set in the **gateway-site.xml** file. If you are unsure whether the default values have been changed, contact your IT department.

   - **Cluster name** - The cluster name as configured in Knox. The default is "Default".
8. In the **HDFS section**, select **WebHDFS**, **HttpFS** or **NFS** as the HDFS access method. If you are accessing MapR, it is recommended to use HttpFS.

   Note When the **Use Knox Gateway** option is selected, the **NameNode**, **HttpFS Host**, and **Port** fields described below are not relevant (and are therefore hidden).

   - If you selected **WebHDFS**:
     - In the **NameNode** field, specify the IP address of the NameNode.

       Note This is the Active node when High Availability is enabled (see below).

     - Replicate supports replication to an HDFS High Availability cluster. In such a configuration, Replicate communicates with the Active node, but switches to the Standby node in the event of failover. To enable this feature, select the **High Availability** check box. Then, specify the FQDN (Fully Qualified Domain Name) of the Standby NameNode in the **Standby NameNode** field.
     - In the **Port** field, optionally change the default port (50070).
     - In the **Target Folder** field, specify where to create the data files on HDFS.
   - If you selected **HttpFS**:
     - In the **HttpFS Host** field, specify the IP address of the HttpFS host.
     - In the **Port** field, optionally change the default port (14000).
     - In the **Target Folder** field, specify where to create the data files on HDFS.

- If you selected **NFS**:
  - In the **Target folder** field, enter the path to the folder located under the MapR cluster mount point. For example: `/mapr/my.cluster.com/data`
  - In order to do this, you first need to mount the MapR cluster using NFS. For information on how to do this, refer to the MapR help.

Select **WebHDFS**, **HttpFS** or **NFS** as the HDFS access method. If you are accessing MapR, it is recommended to use HttpFS.

9. In the **Hive Access section**, do the following:
   a. From the **Access Hive using** drop-down list, select one of the following options:

> **Note** When the **Use Knox Gateway** option is selected, the Host and **Port** fields described below are not relevant (and are therefore hidden).

   - **ODBC** - Select this option to access Hive using an ODBC driver (the default). Then continue with the Host field.
   - **Note:** If you select his option, make sure that the latest 64-bit ODBC driver for your Hadoop distribution is installed on the Attunity Replicate Server machine.
   - **WebHCat** - Select this option to use the WebHCat REST API as the access method. Then continue from Step Setting General Connection Properties.
   - **HQL scripts** - When this option is selected, Replicate will generate HQL table creation scripts in the specified **Script folder.**

> **Note** When this option is selected, the target storage format must be set to "Text".

   - **No Access** - When this option is selected, after the data files are created on HDFS, Replicate will take no further action.

   b. In the **Host** field, specify the IP address of the Hive machine.
   c. In the **Port** field, optionally change the default port.
   d. In the **Database** field, specify the name of the Hive target database.

Setting Advanced Connection Properties

The table below describes the settings in the **Advanced** tab.

Table 9.6 | Hadoop Target - Advanced Properties

Setting	Description
File Format	**Expand this section to specify or view the file format settings.**
Target storage format	Select one of the following target storage formats: Text (the default), Avro, ORC, Parquet, Sequence.

Table 9.6 | Hadoop Target - Advanced Properties (Cont.)

Setting	Description
	Note If **Avro**, **ORC** or **Parquet** is selected or if the target tables have skews/buckets, Replicate first converts the source data to a temporary sequence file and then runs a Hive process to convert the sequence file to the desired target format. **Note** Unlike other binary formats that need to be converted to the desired target format (see above), when **Sequence** format is selected, the data is loaded directly to the target and stored in an external table (in sequence format). Note that Snappy compression is not available for sequence format. See also: Prerequisites for using the Cloudera Distribution as a Hadoop Target.
Use Default SerDe Other SerDe	Choose the SerDe interface to use when accessing the Hive database tables. The default is **LazySimpleSerde**. **LazySimpleSerde** creates the target files in delimited text file format. To create the target files in a different format, select the **Other SerDe** field and then specify the name of the SerDe that you want to use.
Field delimiter	The delimiter that will be used to separate fields in the target file. The default is `\001`. **Note** When using other SerDe: The default name for the field delimiter property is `field.delim`. If you selected **Other SerDe** and the specified SerDe uses a different property name (e.g. `separatorChar`), in addition to specifying the property value here, you also need to specify both the property name **and** its value in the SerDe properties field (e.g. `separatorChar=\t`).
Null value	The value that will be used to indicate a null value in the target file. When using the default SerDe (LazySimpleSerde), setting the null value is supported from Hive 0.13. **Example (where @ is the null value):** mike,male,295678 sara,female,@

Table 9.6 | Hadoop Target - Advanced Properties (Cont.)

Setting	Description
	Note When using other SerDe: The default name for the null value property is `serialization.null.format`. If you selected **Other SerDe** and the specified SerDe uses a different property name (e.g. `nullChar`), in addition to specifying the property value here, you also need to specify both the property name **and** its value in the SerDe properties field (e.g. `nullChar=@`).
Escape character	**When using LazySimpleSerde:** The escape character is used to escape the field delimiter character. When a field delimiter is escaped, it is interpreted as actual data, and not as a field delimiter. **Example (where \ is the escape character and a comma is the field delimiter):** sunroof\,power-steering **When using Other SerDe:** The escape character is used to escape the quote character. **Example (where \ is the escape character and double quotes is the quote character):** "\"sunroof, power-steering\"" **Note** When using other SerDe: The default name for the escape character property is `escape.delim`. If you selected **Other SerDe** and the specified SerDe uses a different property name (e.g. `escapeChar`), in addition to specifying the property value here, you also need to specify both the property name **and** its value in the SerDe properties field (e.g. `escapeChar={`).
Record delimiter	The `\n` delimiter is used to separate records (rows) in the target files. When using the default SerDe (LazySimpleSerde), the record delimiter cannot be changed. **Note** When using other SerDe: The default name for the record delimiter property is `line.delim`. If you selected **Other SerDe** and the specified SerDe uses a different property name (e.g. `recordChar`), in addition to specifying the property value here, you also need to specify both the property

Table 9.6 | Hadoop Target - Advanced Properties (Cont.)

Setting	Description
	name **and** its value in the SerDe properties field (e.g. `recordChar=\r`).
Quote character	The quote character is used to escape the field delimiter character. When a field delimiter is escaped, it is interpreted as actual data, and not as a field delimiter. Note that the quote character is not available when using the default SerDe (LazySimpleSerde). **Example (where double-quotes is the quote character):** "sunroof,power-steering" **Note** When using other SerDe: The default name for the quote character property is `quote.delim`. If you selected **Other SerDe** and the specified SerDe uses a different property name (e.g. `quoteChar`), in addition to specifying the property value here, you also need to specify both the property name **and** its value in the SerDe properties field (e.g. `quoteChar='`).
SerDe properties	Enter the SerDe properties if **Other SerDe** is selected and the SerDe properties are not the same as the Hadoop defaults (`field.delim, serialization.null.format, escape.delim, line.delim, quote.delim`). The properties should be written using the following format: "KEY1=VALUE1,KEY2=VALUE2,KEY3=VALUE3" The list of properties should begin and end with a quotation mark. **Example:** `"separatorChar=\t,escapeChar={,quoteChar='"` **Note** When " is specified as a value, it needs to be enclosed with quotation marks and escaped with a quotation mark, as follows: """"
Add metadata header	You can optionally add a header row to the data files. The header row can contain the source column names and/or the intermediate (i.e. Replicate) data types. Example of a target file with a header row when both **With column names** and **With data types** are selected: Position:DECIMAL(38,0),Color:VARCHAR(10)

Table 9.6 | Hadoop Target - Advanced Properties (Cont.)

Setting	Description
	1,"BLUE" 2,"BROWN" 3,"RED" ... **Note** This option is only available when "No Access" is selected as the Hive access method (in the **General** tab) and the **Target storage format** is "Text".
File Attributes	**Expand this section to specify or view the file attributes.**
Use Hadoop defaults	Select to work with the default block size of your Hadoop target.
Use this block size (MB)	Select to work with a different block size. The default value is 64.
Maximum file size	Specify the maximum file size of each target file. When the data reaches the maximum size, the file will be closed and written to the specified target folder.
Compress files using	Select the compression method to use on HDFS. **Note** Cloudera ODBC drivers 2.5.20 and above do not support the Snappy compression method. **Note** To use Snappy compression when the Setting Advanced Connection Properties is set to **Avro**, **Parquet** or **Text**, you must add the following values to the `hive.security.authorization.sqlstd.confwhitelist.append` property in the Hive configuration file:] » **For Avro:** `|hive.exec.compress.output|avro.output.codec` » **For Parquet:** `|hive.exec.compress.output|parquet.compression` » **For Text:** `|hive.exec.compress.output|parquet.compression` If the value(s) already exist in the `hive.security.authorization.sqlstd.confwhitelist` property, you do not need to add them to the `hive.security.authorization.sqlstd.confwhitelist.append`

Table 9.6 | Hadoop Target - Advanced Properties (Cont.)

Setting	Description
	property. See also: Prerequisites for using the Cloudera Distribution as a Hadoop Target.
Change Processing	**Expand this section to specify or view change processing settings.**
Consider state idle when no changes have been processed for	Specify how long to wait before considering the state to be idle. In idle state, you can create files from data that has already been processed if the specified size and time conditions are met (see below).
File size reaches	Specify the minimum size of the data required to create a file in idle state.
Elapsed time reaches	Specify the maximum time to wait before applying the changes in idle state.

Note To facilitate rapid delivery of DDL messages, files are uploaded immediately, regardless of the specified **File size reaches** or **Elapsed time reaches** values.

Preventing ODBC Connection Timeouts

The default query timeout value is 600 seconds, which should be sufficient for most situations. However, when loading very large tables, you may need to increase the value to prevent timeouts. This can be done using the following internal parameter:

`executeTimeout`

See below for instructions on setting internal parameters.

Internal Parameters

Internal parameters are parameters that are not exposed in the UI. You should only use them if instructed by Attunity Support.

To add internal Attunity Replicate parameters:

1. Click the **Internal Parameters** link.
   The **Internal Parameters** dialog box opens.
2. In the edit box, type the name of the parameter you need to add and then click it.
3. The parameter is added to the table below the search box with its default value.
4. Change the default value as required.

5. To reset the parameter value to its default, click the "Restore default value" icon at the end of the row.

Settings Summary

You can view a summary of your settings by clicking the **Setting Summary** link. This is useful if you need to send a summary of your settings to Attunity Support.

Using Kerberos Authentication

Whether Attunity Replicate Server is running on Linux or Windows, you can configure it to authenticate itself against the Hadoop cluster using Kerberos (See Setting General Connection Properties and Setting General Connection Properties).

This requires you to perform the following steps on the Attunity Replicate machine before starting the Attunity Replicate Server.

To use Kerberos authentication on Linux:

Note The commands described below should be issued under the "Attunity" user or under the user that was selected during the Replicate installation.

1. Obtain a valid TGT (Ticket-Granting Ticket) from the Kerberos KDC (Key Distribution Center) but save the TGT to a non-default cache file. Usually, a keytab file is used to perform non-interactive authentication to Kerberos.

   Command Syntax:

   ```
   kinit -kt [keytab_file] -c [cache_file_name] [principal_name]
   ```

2. This step is only required for the global Kerberos ticket file. Set the Kerberos cache environment variable (for Replicate to use later on).

   To set the environment variable:

   a. Change the working directory to the Replicate "bin" directory by issuing the following command (assumes the default installation path):

      ```
      cd /opt/attunity/replicate/bin
      ```

   b. Stop the Attunity Replicate Server services on the Linux by running:

      ```
      /opt/attunity/replicate/bin/arep.ctl stop
      ```

3. Create a file named `site_arep_login.sh` in the Attunity Replicate bin folder.

   a. Add the following command to the file:

      ```
      export KRB5CCNAME=cache_file_name
      ```

      Example:

      export KRB5CCNAME=/temp/kerberos/global.ticket

   b. Save the file and start the Attunity Replicate Server services as normal (see Starting the Attunity Replicate Server Process).

Now, whenever Attunity Replicate needs to use Kerberos authentication, it will perform the following operations:

- When **Use global Kerberos ticket file** is selected: Replicate will check whether the KRB5CCNAME environment variable is set and, if so, will use the ticket(s) inside the cache file specified by the environment variable.
- When **Use specific Kerberos ticket file** is selected:
  - During design-time (e.g. when selecting tables, testing the connection, etc.), Replicate will use the ticket(s) inside the cache file specified by the KRB5CCNAME environment variable.
  - During runtime, Replicate will use the ticket file specified in the Hadoop endpoint settings.

Note If the ticket in the cache file expires or becomes invalid, repeating the `kinit` command shown in **Step 1** above will write a new TGT to the cache file and allow Attunity Replicate to continue working. This can be done without restarting the Attunity Replicate Server.

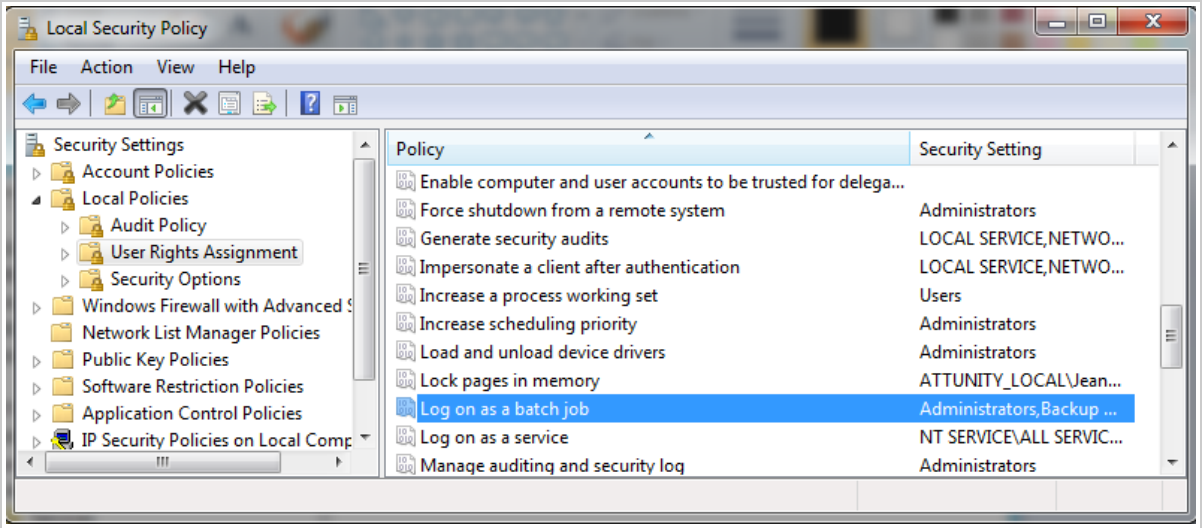
To set Kerberos authentication on Windows:

Before beginning, make sure that the impersonated user (principal) has access to the **Data** directory (<product_dir>**Data**) on the Attunity Replicate server. The **Data** directory contains files (such as CSV files) that are required for the replication task. For Active Directory KDC, the impersonated user is the user configured in the user interface. For MIT KDC, this is the Windows user to which the MIT principal is mapped.

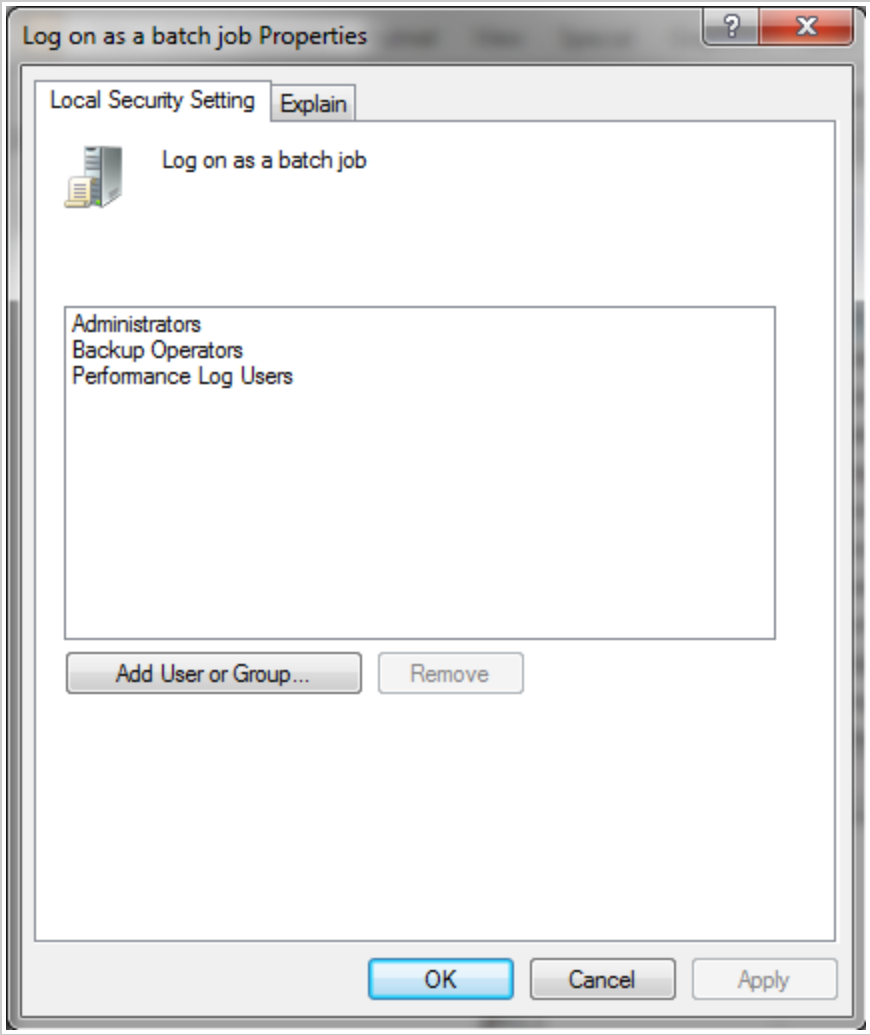
Perform the following steps to ensure that the impersonated user (principal) has the **Log on as a batch job** privilege on the Attunity Replicate server.

These steps are required regardless of the KDC used.

1. On the Attunity Replicate server, open the **Local Security Settings** (**Control Panel > System Security > Administrative Tools > Local Security Policy**).



2. In the console tree, expand **Local Policies** and select **User Rights Assignments**.
3. In the details pane, double-click **Log on as a batch job**.
4. In the **Log on as a batch job Properties** dialog box, on the **Local Security Settings** tab, verify that the respective user is listed. If it is not listed, click **Add User or Group**, then add the user and click **OK**.

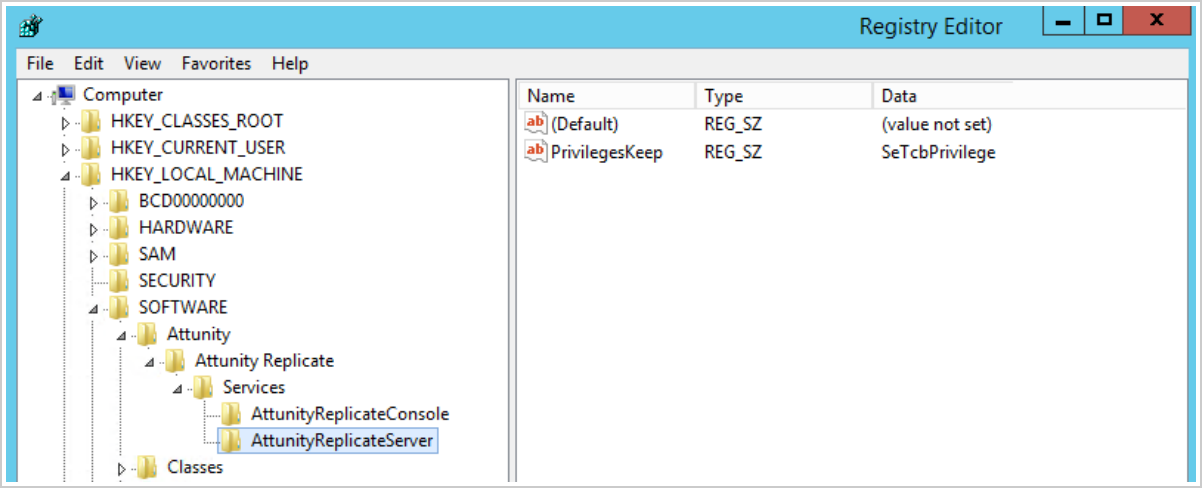


Your changes should take effect immediately.

To set Kerberos authentication on Windows with MIT Kerberos set in one of the endpoints:

If MIT Kerberos is set in one of the endpoints, you need to perform the following steps to allow the Attunity Replicate server process to keep a specific privilege on startup. By default, Attunity Replicate server drops all privileges on startup. These steps are not required if you use Active Directory KDC.

1. Open the Windows registry (regedit.exe).
2. Browse to: HKEY_LOCAL_MACHINE\SOFTWARE\Attunity\Attunity Replicate\Services\AttunityReplicateServer



3. Modify the **`PrivilegesKeep`** string to include the value **`SeTcbPrivilege`**.
4. Close the **Registry Editor** window.
5. Start the **Attunity Replicate Server service**.

Using Teradata Database as a Target

This section describes how to set up and use Teradata Database as a target in a replication task.

In this section:

An Overview of the Teradata Database Target

The Attunity Replicate database for Teradata Database is a powerful operational data warehousing solution that manages Big Data analytics and challenges. Attunity Replicate uses the Teradata Database Parallel Transporter (TPT) API to facilitate data loading. The ODBC API is used for other purposes such as metadata queries (DDL requests) and retrieving information from Teradata Database error tables.

Attunity Replicate for Teradata Database uses the TPT load to bulk load data into a Teradata Database target database. You can replicate data to the Teradata Database from any source database supported by Attunity Replicate. In addition, Attunity Replicate can replicate data from any source database that supports ODBC.

Teradata Database Target Load Options

You can apply changes in one of two modes:

- TPT Stream Mode
- TPT Load Mode

TPT Stream Mode

When using the TPT stream mode, the TPT Stream operator uses the Teradata Database TPump protocol to perform high-speed DML transactions in a near-real-time mode on tables. The TPT STREAM operator is less restrictive than the LOAD operator.

This mode lets tables be queried at the same time that a DML operation takes place.

TPT Load Mode

When using the TPT load mode, the TPT LOAD operator uses the Teradata Database FastLoad protocol to load a large volume of data at high speed into an empty table on the Teradata Database.

The TPT LOAD operator has some restrictions that include the following:

- The target table must be empty.
- The target table cannot have secondary indexes defined.

Database Availability

Teradata Database with the tables that are being used for replication must be available to the system. This can be installed on any computer in your network.

For more information about the requirements for working with Attunity Replicate, see Installation Prerequisites.

Required Teradata Database Software, Environments

The following describes the prerequisites necessary to prepare your environment to work with Attunity Replicate and Teradata Database.

Note Teradata Database must be installed in your network and be reachable from the computer where Attunity Replicate is installed.

Replicate Server for Windows

You must install the following on the same computer where the Attunity Replicate Server is installed:

- Teradata Database ODBC Driver for Windows version 15.00.
- Teradata Database Parallel Processor API (TPT API) with the load and Stream TPT operators. Install either version 14.00 with the latest patch or version 15.10.

Replicate Server for Linux

The following section describes the steps you need to perform to work with Attunity Replicate for Linux and Teradata Database as a target database in a Replicate task. Teradata Database Client requires the DataDirect ODBC driver manager (provided with Teradata Database Client).

Important: A Replicate task cannot be defined with endpoints that use different ODBC Driver Managers. Teradata Database target is accessed using the *DataDirect* ODBC Driver Manager. With the exception of Oracle, Hadoop, File and Replicate Connect sources (which are not subject to the above limitation), all other source endpoints use the *unixODBC* Driver Manager.

To configure a task with a *unixODBC* source and a *DataDirect* target (e.g. Microsoft SQL Server to Teradata Database Target), you need to use the Replicate File Channel. For more information about setting up a task using the File Channel, see Using the Attunity Replicate File Channel.

1. Install Replicate on the Linux machine as described in Attunity Replicate on Linux: Installing, Upgrading and Uninstalling.
2. Install the following Teradata client components:
   - Teradata Database ODBC Driver 14.10 or 15.10 for Linux
   - Teradata Database Parallel Processor API (TPT API) with the Load and Stream TPT operators
3. Copy the **odbc.ini** file to your home directory and rename it to `.odbc.ini`:
   ```
   cp $TD_CLIENT_DIR/odbc_64/odbc.ini ~/.odbc.ini
   ```
4. Open the Teradata odbcinst.ini file:
   ```
   cat $TD_CLIENT_DIR/odbc_64/odbcinst.ini
   ```
   Then verify that it contains a definition for the Teradata ODBC client:
   ```
   [ODBC DRIVERS]
   Teradata=Installed
   [Teradata]
   ```
   Version 14.10: `Driver=/opt/teradata/client/14.10/odbc_64/lib/tdata.so`

   Version 15.10: `Driver=/opt/teradata/client/15.10/lib64/lib/tdata.so`
   ```
   DriverODBCVer=3.51
   ```
5. Check that directory **/usr/lib64** contains symbolic links to the DataDirect driver manager shared libraries:
   ```
   ll /usr/lib64/libodbc*.so
   ```
   The output should look like this (e.g. for version 14.10):
   ```
   lrwxrwxrwx 1 root root 47 Oct 28 14:58 /usr/lib64/libodbcinst.so ->
    /opt/teradata/client/ODBC_64/lib/libodbcinst.so
   lrwxrwxrwx 1 root root 43 Oct 28 14:58 /usr/lib64/libodbc.so ->
    /opt/teradata/client/ODBC_64/lib/libodbc.so
   ```
6. Add the Teradata Database name to the hosts file as described in Editing the Hosts File.
7. Run the following commands:
   ```
   export LD_LIBRARY_PATH=/usr/lib64:$TD_CLIENT_
   DIR/tbuild/lib64:/opt/attunity/replicate/lib
   export AREP_ODBC_DRIVER_MANAGER=/opt/teradata/client/14.10/odbc_
   64/lib/libodbc.so
   export ODBCINI=/opt/teradata/client/14.10/odbc_64/odbc.ini
   ```

Providing Access to the Teradata Database

The Attunity Replicate user who is working with the Teradata Database must be registered as a user in the Teradata Database. This is the user that is entered in the dialog box when Setting General Connection Properties. You must grant Teradata Database access to this user before configuring the database in Attunity Replicate.

Editing the Hosts File

To enable Attunity Replicate to access the Teradata Database, you need to add the Teradata Database machine IP/name and database mappings to the Windows/Linux hosts file.

To add the Teradata Database mappings to the hosts file:

1. Open the Windows/Linux hosts file on the Attunity Replicate machine.

   On Windows, the default path for the hosts file is:

   ```
   ~:\Windows\System32\drivers\etc\hosts
   ```

   On Linux, the default path for the hosts file is:

   ```
   /etc/hosts
   ```

2. Add the following line (note the "`cop1`" after the database name):

   ```
   <Teradata Database IP address/hostname> <Teradata Database name>cop1
   ```

   Example:

   ```
   123.123.123.1 teradatadbonecop1
   ```

   Note Make sure that the database name added to the hosts files is the same as the database specified in the **Default database** field in the Teradata Database target database settings.

3. Save your changes.

Security Requirements

A user must have the following privileges granted in the Teradata Database to use a Teradata Database target in an Attunity Replicate task:

- GRANT SELECT ON <database>
- GRANT INSERT ON <database>
- GRANT DELETE ON <database>
- GRANT UPDATE ON <database>
- GRANT EXECUTE ON <database>
- GRANT EXECUTE FUNCTION ON <database>
- GRANT EXECUTE PROCEDURE ON <database>
- GRANT CREATE TABLE ON <database>
- GRANT DROP TABLE ON <database>

- GRANT CREATE VIEW ON <database>
- GRANT DROP VIEW ON <database>
- GRANT NONTEMPORAL on <database>
- GRANT CHECKPOINT ON <database>

  When the Stream TPT Operator is selected (in the **Advanced** tab), the following privilege is also required:
- GRANT CREATE MACRO ON <database>

Teradata Database Data Types

The Teradata Database database for Attunity Replicate supports most Teradata Database data types. The following table shows the Teradata Database target data types that are supported when using Attunity Replicate and the default mapping from Attunity Replicate data types. Unsupported data types are listed below the table.

Note Teradata Database does not support applying changes to binary data types in **Batch optimized apply** mode. For more information on **Batch optimized apply** mode, see Changes Processing Tuning.

For information on how to view the data type that is mapped from the source, see the section for the source database you are using. For additional information about Attunity Replicate data types, see Replicate Data Types.

Table 9.7 | Supported Teradata Database Data Types with Mapping from Attunity Replicate Data Types

Attunity Replicate Data Types	Teradata Database Data Types
BOOLEAN	**BYTEINT**
BYTES	**VARBYTE** (Size) **Note** Maximum size is 640000.
DATE	**DATE**
TIME	**TIME (P)** Precision is not included unless it is less than 0.
DATETIME	**TIMESTAMP (P)**

Table 9.7 | Supported Teradata Database Data Types with Mapping from Attunity Replicate Data Types (Cont.)

Attunity Replicate Data Types	Teradata Database Data Types
	Precision is not included unless it is less than 0.
INT1	**BYTEINT**
INT2	**SMALLINT**
INT4	**INTEGER**
INT8	**BIGINT**
NUMERIC	**NUMERIC** (P, S) Scale is not included unless it is less than 0.
REAL4	**FLOAT** FLOAT is equivalent to REAL and DOUBLE PRECISION.
REAL8	**FLOAT**
STRING See also the note in Teradata Database Data Types below.	**VARCHAR** (Size) **Note**: Maximum size is 64000.
UINT1	**BYTEINT**
UINT2	**SMALLINT**
UINT4	**INTEGER**
UINT8	**BIGINT**
WSTRING See also the note in Teradata Database Data Types below.	**VARCHAR** (Size) **Note** Maximum size is 640000.

Table 9.7 | Supported Teradata Database Data Types with Mapping from Attunity Replicate Data Types (Cont.)

Attunity Replicate Data Types	Teradata Database Data Types
Note About Teradata Database LOB support: Full LOB data types are not supported in the Teradata Database. For information on including Limited-size LOB data types in the replication, see Metadata. Note also that the size of a row in the Teradata Database *cannot exceed 64KB*. This should be taken into consideration when specifying the maximum LOB size in the **Metadata** tab. See also the note in CLOB below.	
BLOB	**VARBYTE (${MAX_LOB_SIZE})** MAX_LOB_SIZE is the maximum LOB size specified in Limited-Size LOB Mode.
CLOB **Note** By default, Replicate multiplies the value of each `varchar` column by three, in order to support NLS. For example, a `varchar` column with 36 characters in the source database will have 108 characters in Teradata Database. This may result in Teradata Database `varchar`columns being longer than you actually need them (and unnecessarily increasing the row size). In such cases, you can override the default multiplication factor by using the `nlsFactor` internal parameter. For instructions on using the `nlsFactor` parameter, contact Attunity Support.	**VARCHAR (${MAX_LOB_SIZE})** Unicode case-insensitive character set. MAX_LOB_SIZE is the maximum LOB size specified in Limited-Size LOB Mode.
NCLOB See the note in CLOB above.	**VARCHAR (${MAX_LOB_SIZE})** Case-insensitive character set. MAX_LOB_SIZE is the maximum LOB size specified in Limited-Size LOB Mode.

The following Teradata Database data types are not supported:

- PERIOD

Setting General Connection Properties

This section describes how to configure general connection properties. For an explanation of how to configure advanced connection properties, see **Setting Advanced Connection Properties** below.

To add a Teradata Database Target to Attunity Replicate:

1. In **Tasks** view, click **Manage Endpoint Connections** to open the **Manage Endpoints Connections** dialog box. Then click the **New Endpoint Connection** button. For more information on adding an endpoint to Attunity Replicate, see Working with Endpoints.
2. In the **Name** field, type a name for your database. This can be any name that will help to identify the database being used.
3. In the **Description** field, type a description that helps to identify the Teradata Database. This is optional.
4. Select **TARGET** as the database **role**.
5. Select **Teradata Database** as the database **Type**.
6. Type the **Server** name. This is the name of the computer with the Teradata Database instance you want to work with.
7. Type the Teradata Database authentication information (**Username**, **Password**) for the authorized user for this Teradata Database. If you do not know this information, see your Teradata Database system manager.

Note Consider the following:

- If you are using the **Advanced** tab to create a custom string, make sure to include the **USERNAME** property. A **Password** can also be included but is not required. See Setting Advanced Connection Properties for more information.
- This information is case sensitive.
- If you want to set custom properties for this database, see Setting Advanced Connection Properties.
- To determine if you are connected to the database you want to use or if the connection information you entered is correct, click **Test Connection**.

  If the connection is successful a message in green is displayed. If the connection fails, an error message is displayed at the bottom of the dialog box.

  To view the log entry if the connection fails, click **View Log**. The server log is displayed with the information for the connection failure. Note that this button is not available unless the test connection fails.

Important: Make sure that the Teradata Database user entered in the Teradata

Database Authentication section has the correct access privileges. For information on how to provide the required privileges, see Security Requirements.

8. Type the **Default database** name or select one from the list of available endpoints. This is the name of the Teradata Database where you are replicating the data to.

   For more information, see Teradata Database Target Load Options.

Setting Advanced Connection Properties

You can set custom properties or change the default settings for various parameters by adding them to a custom connect string in the **Advanced** tab of the Add Database dialog box.

You can set the following parameters:

- **TPT Operator**: Select the TPT Operator used to access the Teradata Database. The possible options are:
  - **Load**: Select this to use the TPT Load Mode.
  - **Stream**: Select this to use the TPT Stream Mode.

  See Teradata Database Target Load Options.
- TPT Attributes: You can define one or more of the following attributes:
- **Account String**: The account (database server login) that the DBA assigned to the user-name for the Attunity Replicate user.
- **Buffer Size**: The output buffer size (in KB) for sending Load parcels to the Teradata Database.

  You can enter a value from 1 to 64.
- **Buffers**: The number of request buffers used.

  You can enter any value from 2 or higher.
- **Explicit sessions range**: Select this if you want set a minimum and/or maximum number of sessions that can log on to the Teradata Database.
  - **Maximum**: The maximum number of sessions that can log on to the Teradata Database. The default value is 1. The value cannot be higher than the number of Access Module Processors (AMPs) available.
  - **Minimum**: The minimum number of sessions that can log on to the Teradata Database.
- **Dynamic statement packing**: Select this check box if you want the stream driver to dynamically determine the maximum possible pack for the current STREAM job.
  - **Statement packing**: Use the counter or type the number of statements that can be packed into a multiple-statement request (STREAM).

    You can enter a value from 2 to 600.

    This is available only if **Statement packing** is not selected.
- **Additional ODBC connection properties**: Type any additional ODBC connection properties, if required.

Internal Parameters

Internal parameters are parameters that are not exposed in the UI. You should only use them if instructed by Attunity Support.

To add internal Attunity Replicate parameters:

1. Click the **Internal Parameters** link.

   The **Internal Parameters** dialog box opens.
2. In the edit box, type the name of the parameter you need to add and then click it.
3. The parameter is added to the table below the search box with its default value.
4. Change the default value as required.
5. To reset the parameter value to its default, click the "Restore default value" icon at the end of the row.

Settings Summary

You can view a summary of your settings by clicking the **Setting Summary** link. This is useful if you need to send a summary of your settings to Attunity Support.

Using PostgreSQL as a Target

This section describes how to set up and use PostgreSQL database as a target in a replication task.

In this section:

Prerequisites

The following section describes the client prerequisites when replicating to a PostgreSQL target.

- **Attunity Replicate Server for Windows:**
  - **The PostgreSQL ODBC Driver:** The PostgreSQL ODBC Driver `psqlodbc_09_03_0300-x64-1` or above must be installed on the Attunity Replicate machine.
  - **pgAdmin:** The pgAdmin Open Source administration and development platform for PostgreSQL must be installed on the Attunity Replicate machine.

    Note Make sure that the psql.exe path (e.g. "C:\Program Files\PostgreSQL\9.6\bin") is added to the system PATH.

    Note If PgAdmin and the Greenplum client are installed on the same Replicate Server, tasks configured to use a PostgreSQL target will fail.

- **Attunity Replicate Server for Linux**: On the Attunity Replicate machine:
  - Install `postgresql94-9.4.4-1PGDG.<OS Version>.x86_64.rpm`. This is the package that contains the psql executable.

    For example, `postgresql94-9.4.4-1PGDG.rhel7.x86_64.rpm` is the package required for Red Hat 7.
  - Install unixODBC driver `postgresql94-odbc-09.03.0400-1PGDG.<OS version>.x86_64` or above for Linux, where `<OS version>` is the OS of the Replicate Server machine.

    For example, `postgresql94-odbc-09.03.0400-1PGDG.<rhel7>.x86_64` is the client required for Red Hat 7.
  - Makes sure that the `/etc/odbcinst.ini` file contains an entry for PostgreSQL, as in the following example:

```
[PostgreSQL]
Description = PostgreSQL ODBC driver
Driver = /usr/lib/odbc/psqlodbca.so
Setup = /usr/lib/odbc/libodbcpsqlS.so
Debug = 0
CommLog = 1
UsageCount = 2
```

Security Requirements

The user specified in the **General** tab when Setting General Connection Properties must be a registered user in the PostgreSQL database.

PostgreSQL Database Target Data Types

The PostgreSQL endpoint for Attunity Replicate supports most PostgreSQL database data types. The following table shows the PostgreSQL database target data types that are supported when using Attunity Replicate and the default mapping from Attunity Replicate data types. Unsupported data types are listed below the table.

For information on how to view the data type that is mapped from the source, see the section for the source database you are using. For additional information about Attunity Replicate data types, see Replicate Data Types.

Table 9.8 | Supported PostgreSQL database Data Types with Mapping from Attunity Replicate Data Types

Attunity Replicate Data Types	PostgreSQL database Data Types
BOOL	**BOOL**
BYTES	**BYTEA**
DATE	**DATE**
TIME	**TIME**
DATETIME	If scale is => 0 and =< 6, then: **TIMESTAMP** If scale is => 7 and =< 9, then: **VARCHAR** (37)
INT1	**SMALLINT**
INT2	**SMALLINT**
INT4	**INTEGER**

Table 9.8 | Supported PostgreSQL database Data Types with Mapping from Attunity Replicate Data Types (Cont.)

Attunity Replicate Data Types	PostgreSQL database Data Types
INT8	**BIGINT**
NUMERIC	**DECIMAL** (P, S)
REAL4	**FLOAT**4
REAL8	**FLOAT8**
STRING	If length is 1 - 21845, then: **VARCHAR** (Length in Bytes = The STRING value multiplied by three) If length is 21846 - 2147483647, then: **VARCHAR** (65535)
UINT1	**SMALLINT**
UINT2	**INTEGER**
UINT4	**BIGINT**
UINT8	**BIGINT**
WSTRING	If length is 1 - 21845, then: **VARCHAR** (Length in Bytes = The WSTRING value multiplied by three) If length is 21846 - 2147483647, then: **VARCHAR** (65535)
BLOB	**BYTEA**
NCLOB	**TEXT**
CLOB	**TEXT**

Data Types when Replicating from a PostgreSQL Source

When replicating from a PostgreSQL source, the target table will be created with the same data types for all columns, apart from columns with user-defined data types. In such cases, the data type will be created as "character varying" in the target.

Setting General Connection Properties

This section describes how to configure general connection properties. For an explanation of how to configure advanced connection properties, see **Setting Advanced Connection Properties** below.

To add a PostgreSQL target endpoint to Attunity Replicate:

1. In **Tasks** view, click **Manage Endpoint Connections** to open the **Manage Endpoints Connections** dialog box. Then click the **New Endpoint Connection** button. For more information on adding an endpoint to Attunity Replicate, see Working with Endpoints.
2. In the **Name** field, type a name for your database. This can be any name that will help to identify the database being used.
3. In the **Description** field, type a description that helps to identify the PostgreSQL database. This is optional.
4. Select **TARGET** as the database **role**.
5. Select **PostgreSQL** as the database **Type**.
6. Type the **Server** name. This is the name or IP address of the computer with the PostgreSQL database that you want to access.
7. Optionally, change the default port (5432).
8. Enter the PostgreSQL database authentication information (**User name**, **Password**) of an authorized PostgreSQL user. If you do not know this information, see your PostgreSQL database system manager.

   Note Consider the following:
   - This information is case sensitive.
   - To determine if you are connected to the database you want to use or if the connection information you entered is correct, click **Test Connection**.

     If the connection is successful a message in green is displayed. If the connection fails, an error message is displayed at the bottom of the dialog box.

     To view the log entry if the connection fails, click **View Log**. The server log is displayed with the information for the connection failure. Note that this button is not available unless the test connection fails.

   Important: Make sure that the specified PostgreSQL database user has the correct access privileges.

9. Type the **database name** or select one from the list of available endpoints. This is the name of the PostgreSQL database to which you are replicating data.
10. Click **OK** to save your settings and close the dialog box.

Setting Advanced Connection Properties

In the **Advanced** tab, you can set the following properties:

You can set the following properties:

- **Max file size (KB)**: Select or type the maximum size (in KB) of a CSV file before the file is loaded into the PostgreSQL target database. The default value is 32000 KB.

Internal Parameters

Internal parameters are parameters that are not exposed in the UI. You should only use them if instructed by Attunity Support.

To add internal Attunity Replicate parameters:

1. Click the **Internal Parameters** link.

   The **Internal Parameters** dialog box opens.
2. In the edit box, type the name of the parameter you need to add and then click it.
3. The parameter is added to the table below the search box with its default value.
4. Change the default value as required.
5. To reset the parameter value to its default, click the "Restore default value" icon at the end of the row.

Settings Summary

You can view a summary of your settings by clicking the **Setting Summary** link. This is useful if you need to send a summary of your settings to Attunity Support.

Using a File as a Target

This section describes how to set up and use delimited text files as a target in a replication task. You can use the File target endpoint to export database tables to files, which can then be used as a source in a Replicate task with a File source endpoint.

In this section:

File Target Overview

When using a File as a target in a Replicate task, both the Full Load and the CDC data are written to CSV or JSON files (depending on the endpoint settings). While the explanations in this topic relate to CSV files, the same is true for JSON files.

Full Load files are named using incremental counters e.g. LOAD00001.csv, LOAD 00002.csv, etc. whereas Apply Changes files are named using timestamps e.g. 20141029-1134010000.csv.

Note

- The Apply Changes CSV files appear with a **.tmp** extension while they are in idle state. For more information on idle state, see Change Processing.
- When the Create metadata files in the target folder option is enabled, a corresponding metadata file is created using the same naming format, but with a **.dfm** extension.

For each source table, a folder is created under the specified target folder. All files - i.e. Full Load, Apply Changes, and Metadata (if enabled) - are written to the relevant folder, according to the settings defined in the File target's **General** tab.

After a task completes, you can define another task with a File source endpoint that uses the generated CSV files.

DDL Handling

When a DDL change is captured, Replicate will close the data file and also create a DFM file if the **Create metadata files in the target folder** option is enabled. When the next batch of changes arrives, Replicate will create a new data file containing the changes. Note that the DFM file created for the new data file will match the new table structure.

Limitations

The following limitations apply to the File target endpoint:

- Only the following DDLs are supported: Truncate table, Drop table, Create table, Add Column, Rename Column, Drop Column, and Convert Data Type.
- Full LOB Mode is not supported
- UPDATE and DELETE statements are not supported in Apply Changes replication mode
- Batch Optimized Apply mode is not supported
- Target lookup is not supported
- The <target folder> parameter cannot include special characters

File Target Data Types

The following table shows the default mapping from Attunity Replicate data types to File target data types. Note that the data type mapping is only relevant if the Create metadata files in the target folder option is enabled.

For information on source data type mappings, see the section for the source endpoint you are using.

For additional information about Attunity Replicate data types, see Replicate Data Types.

Table 9.9 | Supported File Target Data Types with Mapping from Attunity Replicate Data Types

Attunity Replicate Data Types	File Target Data Types
DATE	DATE
TIME	TIME
DATETIME	DATETIME
BYTES	BYTES (length)
BLOB	BLOB
REAL4	REAL4 (7)
REAL8	REAL8 (14)
INT1	INT1 (3)
INT2	INT2 (5)
INT4	INT4 (10)
INT8	INT8 (19)
UINT1	UINT1 (3)
UINT2	UINT2 (5)

Table 9.9 | Supported File Target Data Types with Mapping from Attunity Replicate Data Types (Cont.)

Attunity Replicate Data Types	File Target Data Types
UINT4	UINT4 (10)
UINT8	UINT8 (20)
NUMERIC	NUMERIC (p,s)
STRING	STRING (Length)
WSTRING	STRING (Length)
CLOB	CLOB
NCLOB	NCLOB
BOOLEAN	BOOLEAN (1)

Setting General Properties

This section describes how to configure general properties. For an explanation of how to configure advanced properties, see **Setting Advanced Properties** below.

To configure general properties for the File target endpoint:

1. In **Tasks** view, click **Manage Endpoint Connections** to open the **Manage Endpoints Connections** dialog box. Then click the **New Endpoint Connection** button. For more information on adding an endpoint to Attunity Replicate, see Working with Endpoints.
2. In the **Name** field, type a name for your endpoint. This can be any name that will help to identify the endpoint being used.
3. In the **Description** field, type a description that helps to identify the File endpoint. This is optional.
4. Select **TARGET** as the endpoint **role**.
5. Select **File** as the endpoint **Type**.
6. In the **Target folder** field, specify the full path of the folder to which you the target files to be written.
7. Configure the remaining settings in the **General** tab as described in the following table.

Table 9.10 | File Target Endpoint - General Tab Options

Option	Description
File Format	
Delimiters can be standard characters or a hexadecimal (hex) value. Note that the "`0x`" prefix must be used to denote a hexadecimal delimiter (e.g. `0x01 = SOH`). In the **Field delimiter**, **Record delimiter** and **Null value** fields, the delimiter can consist of con-	

Table 9.10 | File Target Endpoint - General Tab Options (Cont.)

Option	Description
catenated hex values (e.g. `0x0102` = `SOHSTX`), whereas in the **Quote character** and **Escape character** fields, it can only be a single hex value. **Note** The hexadecimal number `0x00` is not supported (i.e. only `0x01-0xFF` are supported).	
Format	You can choose to create the target files in **CSV** or **JSON** format. In a JSON file, each record is represented by a single line, as in the following example: { "book_id": 123, "title": "Alice in Wonderland", "price": 6.99, "is_hardcover": false } { "book_id": 456, "title": "Winnie the Pooh", "price": 6.49, "is_hardcover": true } { "book_id": 789, "title": "The Cat in the Hat", "price": 7.23, "is_hardcover": true } **Note** If you choose **JSON** format , the following fields will be hidden as they are only relevant to CSV format: **Field delimiter**, **Record delimiter**, **Null value**, **Quote character**, **Escape character**, **Code page**, and **Add metadata header**.
Field Delimiter	The delimiter that will be used to separate columns in the target files. The default is a comma. **Example using a comma as a delimiter:** mike,male
Record delimiter	The delimiter that will be used to separate records (rows) in the target files. The default is a newline (`\n`). **Example using an asterisk as a delimiter:** mike,male*sara,female
Null value	The string that will be used to indicate a null value in the target files. **Example (where * is the row delimiter and @ is the null value):** mike,male,295678*sara,female,@
Quote character	The character that will be used at the beginning and end of a column. The default is the double-quote character ("). When a column that contains column delimiters is enclosed in double-quotes, the column delimiter characters are interpreted as actual data, and not as column delimiters. **Example (where a comma is the column delimiter):** "sunroof, power-steering"

Table 9.10 | File Target Endpoint - General Tab Options (Cont.)

Option	Description
Escape character	The character used to escape a quote character in the actual data. **Example (where" is the quote character and \ is the escape character):** 1955,"old, \"rare\", Chevrolet",$1000
Code page	Specify the code page of your target files if it is different from the default (65001). **Note** Windows and Linux systems use different code page conventions. The specified code page must comply with the code page convention of the source file system.
Add metadata header	You can optionally add a header row to the data files. The header row can contain the source column names and/or the intermediate (i.e. Replicate) data types. Example of a target file with a header row when both **With column names** and **With data types** are selected: Position:DECIMAL(38,0),Color:VARCHAR(10) 1,"BLUE" 2,"BROWN" 3,"RED" ...
File Attributes	
Maximum file size	The maximum size a file can reach before it is closed (and optionally compressed). This value applies both to data files and to Reference Files. For information on generating reference files, see Generating Reference Files.
Compress files using	Choose **GZIP** to compress the target files or **NONE** (the default) to leave them uncompressed.
Change Processing	
Consider state idle when no changes have been processed for	Specify how long to wait before considering the state to be idle. In idle state, you can apply changes to files using data that has already been processed if the specified size and time conditions are met (see below).

Table 9.10 | File Target Endpoint - General Tab Options (Cont.)

Option	Description
File size reaches	Specify the maximum size of the data required in order to apply changes to the target file in idle state.
Elapsed time reaches	Specify the maximum time to wait before applying the changes in idle state.
Allow a single transaction to be split into multiple files	By default, a single transaction will **not** be split across multiple files, regardless of the values specified in the **File size reaches** and **Elapsed time reaches** fields. This is important for organizations who require files to contain transactions in their entirety. However, this may also result in very large file sizes. For example, if the **File size reaches** value is **32 MB** and Replicate starts to apply changes for a new **2 GB** transaction at **31 MB**, the target file will only be closed at **2.031 GB**. You should therefore select this option if it is critical that the values in the **File size reaches** and **Elapsed time reaches** fields are adhered to (even if it means splitting a transaction across multiple files).
Metadata Files	
Create metadata files in the target folder	When this option is selected, for each data file, a matching metadata file with a **.dfm** extension will be created under the specified target folder. The metadata files (which are in standard JSON format) provide additional information about the task/data such as the source endpoint type, the source table name, the number of records in the data file, and so on. For a full description of the metadata file as well as possible uses, see Metadata File Description .

Note To determine if you are connected to the endpoint you want to use or if the connection information you entered is correct, click **Test Connection**.

If the connection is successful a message in green is displayed. If the connection fails, an error message is displayed at the bottom of the dialog box.

To view the log entry if the connection fails, click **View Log**. The server log is displayed with the information for the connection failure. Note that this button is not available unless the test connection fails.

Setting Advanced Properties

In the **Advanced** tab, you can enable the creation of a reference file and set post-processing actions. These options are described in detail below.

Table 9.11 | File Target Endpoint - Advanced Tab Options

Option	Description
Generate reference files	Select this option to generate a Reference File containing the full path to the Apply Changes data files. **Note** The reference file only points to the location of the Apply Changes files, and not the Full Load files. For more information on this feature, see Generating Reference Files. For information on using reference files with the File source endpoint, see Reference Files.
Reference file folder	The folder on the Replicate machine in which the Reference File will be created. **Note** Multiple tasks with file target endpoints that have the same target directory are not supported (as each task will attempt to write to the same reference file). **Example:** c:\temp\
Post-process files	You can process the final target files using a custom command. The command will be run whenever a data file is created. **Note** If the **Generate a reference file** option is selected, a row (specifying the file's location) will be added to the Reference File only *after* the command completes successfully. » **Command name** - The location of the command e.g. `C:\utils\move.exe`. » **Working directory** - The directory where you want the command to run. » **Parameters** - Specify any parameters that need to be passed to the command during runtime. You can use the following built-in parameters: `${FILENAME}` - The full path to the CSV file containing the full load or CDC data. `${METADATA_FILENAME}` - The full path to the DFM file containing the metadata. **Note** If the CSV/DFM file paths contain spaces, you must enclose

Table 9.11 | File Target Endpoint - Advanced Tab Options (Cont.)

Option	Description
	these parameters with quotation marks (e.g `"${FILENAME}"`).
	For information on creating metadata files, see Setting General Properties.
	Standard Post Command Exit Codes
	The post-processing command must return a proper exit code. You can either use the standard exit code values described below or set a custom exit code value as described in Setting Post Command Exit Codes with an Internal Parameter.
	» **0** - Success
	» **1** - Recoverable error. The task will recover from the point of failure according to the settings in the Environmental Errors tab.
	» **2** - Table error. If a table error occurs, Replicate will handle the error according to the settings in the Table Errors tab.
	» **3** (or any other value e.g. -100) - Fatal error. The task will fail and not attempt recovery.
	Setting Post Command Exit Codes with an Internal Parameter
	You can use internal parameters to set exit codes with custom values. This is especially useful if your application already uses the standard exit code values.
	See Standard Post Command Exit Codes above for a description of the exit codes.
	» `successExitCode`
	» `recoverableErrorExitCode`
	» `tableErrorExitCode`
	» `fatalErrorExitCode`
	For instructions on setting internal parameters, see Internal Parameters.
After post processing completes	You can decide what to do with the original target files after post-processing completes:
	» **Do nothing** - Leaves the files in their original location
	» **Delete files** - Deletes the files from the disk
	» **Replace file extension with** - Replaces the file extension with the specified extension.

Internal Parameters

Internal parameters are parameters that are not exposed in the UI and should only be used if instructed by Attunity Support.

To add internal Attunity Replicate parameters:

1. Click the **Internal Parameters** link.
   The **Internal Parameters** dialog box opens.
2. In the edit box, type the name of the parameter you need to add and then click it.
3. The parameter is added to the table below the search box with its default value.
4. Change the default value as required.
5. To reset the parameter value to its default, click the "Restore default value" icon at the end of the row.

Settings Summary

You can view a summary of your setting by clicking the View Setting Summary link. This is useful if you need to send a summary of your setting to Attunity Support.

Generating Reference Files

In the **Advanced** tab of the File Target endpoint, you can enable the **Generate a reference file** option. The Reference File contains a list of the Change File locations and is therefore only relevant if the task's Apply Changes or Store Changes options are enabled.

The reference file name format is as follows:

```
<file_target_endpoint_display_name><counter>.csv|json
```

Example:

```
FileTarget00000001.csv
```

Note The counter suffix increases incrementally each time a new Reference File is generated (i.e. when the file reaches the maximum size defined in the **General** tab). Once a new Reference File has been generated, you can delete the old reference file(s) if required.

Whenever an Apply Changes data file is created, a new row is added to the Reference File in the following format:

```
<Source_Table_Name>,<full_path_to_data_file>
```

Example:

```
MyTable,c:\temp\filetarget\dbo.MyTable\20170102-091759447.csv
```

Note that if the **Post-process files** option in the **Advanced** tab is also enabled, the Reference File will be generated *after* the post-processing completes.

Note

- When both the **Post-process files** and the **Delete files** (after post-processing completes) options are enabled, the reference file will not be generated.
- If the **Archive files to folder** (after post-processing completes) option is selected, the reference file will be updated to reflect the archive location of the data files.

Using SAP Sybase IQ as a Target

This section describes how to set up and use a SAP Sybase IQ database as a target database in a replication task.

In this section:

Prerequisites

Make sure the following prerequisites have been met:

- Attunity Replicate is installed on any Windows computer in your network.
- A Sybase account with the required access privileges exists.
- SAP Sybase IQ 64-bit ODBC client installed on the computer where Attunity Replicate is located.

Limitations

The following limitations apply:

- Full LOB mode is not supported.
- Replication of LOBs during Change Processing is not supported in Bulk Apply mode (LOB values are replicated to NULL).

Security Requirements

You must provide SAP Sybase IQ account access to the Attunity Replicate user. This user must have read/write privileges in the SAP Sybase IQ database.

SAP Sybase IQ Target Data Types

The following table shows the Sybase target data types that are supported when using Attunity Replicate and the default mapping from Attunity Replicate data types.

Note SAP Sybase IQ does not support applying changes to binary data types in **Batch optimized apply** mode. For more information on **Batch optimized apply** mode, see Changes Processing Tuning.

For information on how to view the data type that is mapped from the source, see the section for the source database you are using.

For additional information about Attunity Replicate data types, see Replicate Data Types.

Table 9.12 | Supported SAP Sybase IQ Data Types with Mapping from Attunity Replicate Data Types

Attunity Replicate Data Types	SAP Sybase IQ Data Types
BOOLEAN	BIT
BYTES	VARBINARY (Length)
DATE	DATE
TIME	TIME
DATETIME	If scale is => 0 and =< 6, then: TIMESTAMP If scale is => 7 and =< 9, then: VARCHAR (37)
INT1	TINYINT
INT2	SMALLINT
INT4	INTEGER
INT8	BIGINT
NUMERIC	NUMERIC (p,s)
REAL4	REAL
REAL8	DOUBLE
STRING	VARCHAR (Length)
UINT1	TINYINT
UINT2	SMALLINT
UINT4	INTEGER
UINT8	BIGINT
WSTRING	VARCHAR (Length)

Note The SAP Sybase IQ database requires a special license to support LOBs.

BLOB	BLOB
CLOB	CLOB
NCLOB	CLOB

Setting General Connection Properties

This section describes how to configure general connection properties. For an explanation of how to configure advanced connection properties, see **Setting Advanced Connection Properties** below.

Note

- Sybase can also be used as a source database. For information on using Sybase as a source, see Using SAP Sybase ASE as a Target.
- You can also use Sybase files as a source or target. For more information, see Using the Attunity Replicate File Channel.

To add a SAP Sybase IQ database to Attunity Replicate:

1. In **Tasks** view, click **Manage Endpoint Connections** to open the **Manage Endpoints Connections** dialog box. Then click the **New Endpoint Connection** button.
2. In the **Name** field, type a name for your database. This can be any name that will help to identify the database being used.
3. In the **Description** field, type a description that helps to identify the Sybase database. This is optional.
4. Select **TARGET** as the database **role**.
5. Select **SAP Sybase IQ** as the database **Type**.
6. In the **Host** field, enter the hostname or IP address of the computer on which SAP Sybase IQ is installed.
7. In the **Server** field, enter the name of the Sybase server.
8. Optionally, change the default port (7878).
9. Type the Sybase authentication information (**User Name**, **Password**) for the authorized user for this Sybase database. If you do not know this information, see your Sybase database Administrator (DBA).

Note

- This information is required. If you are using the **Advanced** tab to create a custom string, make sure to include the **User Name** and **Password** properties. See Setting Advanced Connection Properties for more information.
- This information is case sensitive.
- If you want to set custom properties for this database, see Setting Advanced Connection Properties.

Important: Make sure that the Sybase user entered in the Sybase Authentication section has the correct access privileges. For information on how to provide the

required privileges, see Security Requirements.

10. In the **database** field, enter the name of the SAP Sybase IQ database.

Setting Advanced Connection Properties

In the **Advanced** tab, you can set the following parameters:

- **Max file size**: Select or type the maximum size (in KB) of a CSV file before the file is loaded into the Sybase database. The default value is 32000 KB.
- **Work in version 12 compatibility mode**: You must enable this option when SAP Sybase IQ version12.x is the replication target. When this option is enabled, you also need to create a shared folder (for the CSV files) that is accessible from the Attunity Replicate Server machine and from the SAP Sybase IQ machine.
  - **Local shared folder path:** The path to the shared folder from the local (Attunity Replicate Server) machine.

    Example: C:\Shared
  - **Remote shared folder path:** The path to the shared folder from the remote (SAP Sybase IQ) machine.

    Example: /mnt/shared

Internal Parameters

Internal parameters are parameters that are not exposed in the UI. You should only use them if instructed by Attunity Support.

To add internal Attunity Replicate parameters:

1. Click the **Internal Parameters** link.

   The **Internal Parameters** dialog box opens.
2. In the edit box, type the name of the parameter you need to add and then click it.
3. The parameter is added to the table below the search box with its default value.
4. Change the default value as required.
5. To reset the parameter value to its default, click the "Restore default value" icon at the end of the row.

Settings Summary

You can view a summary of your settings by clicking the **Setting Summary** link. This is useful if you need to send a summary of your settings to Attunity Support.

Using Pivotal Greenplum as a Target

This section describes how to set up and use a Pivotal Greenplum database as a target in a replication task.

In this section:

An Overview of the Pivotal Greenplum Target

The Attunity Replicate database for Pivotal Greenplum is a powerful operational data warehousing solution that manages Big Data analytics and challenges. Attunity Replicate uses Pivotal Greenplum's Scatter/Gather Streaming technology to help with data integration. This technology handles large amounts of data well.

The Attunity Replicate Pivotal Greenplum database makes it possible to load data from other heterogeneous data sources and maintain the most up to date information. This is done by capturing changes and streaming the changes to the Pivotal Greenplum data warehouse. This can be done with a very low impact on the source data.

The Attunity Replicate Pivotal Greenplum database provides full automation for:

- Schema generation and data type mapping
- Full load of source database tables
- Incremental load of changes made to source tables
- Application of schema changes (DDL) made to the source tables.
- Synchronization between full load and CDC processes.

Manual control is also available if needed.

The Attunity Replicate Pivotal Greenplum database integrates with the Pivotal Greenplum database in two ways:

- **Pivotal Greenplum ODBC API**. This is used for metadata management. The Pivotal Greenplum ODBC API lets Attunity Replicate test the database connection, get the table list and the table schema, build procedures that create external tables to process a file, and invoke the procedures that load the destination table or apply changes from the external table. During the schema generation, data types can be mapped, such as Pivotal Greenplum to Postgres. Primary keys and distribution clauses are generated based on the primary key.
- **Pivotal Greenplum Parallel File Distribution Server (gpfdist)**. This utility is used with read-only external tables for fast, parallel data loading into a Pivotal Greenplum data warehouse. gpfdist uses maximum parallelism while reading from external tables.

  Attunity Replicate works closely with gpfdist to take advantage of its optimized fast, parallel loading facilities. Attunity Replicate uses the Pivotal Greenplum Parallel File Distribution Server to support both full load and incremental load activities.

See Attunity Replicate Pivotal Greenplum database Architecture Overview for a description of the system architecture used with the Pivotal Greenplum database.

Attunity Replicate Pivotal Greenplum database Architecture Overview

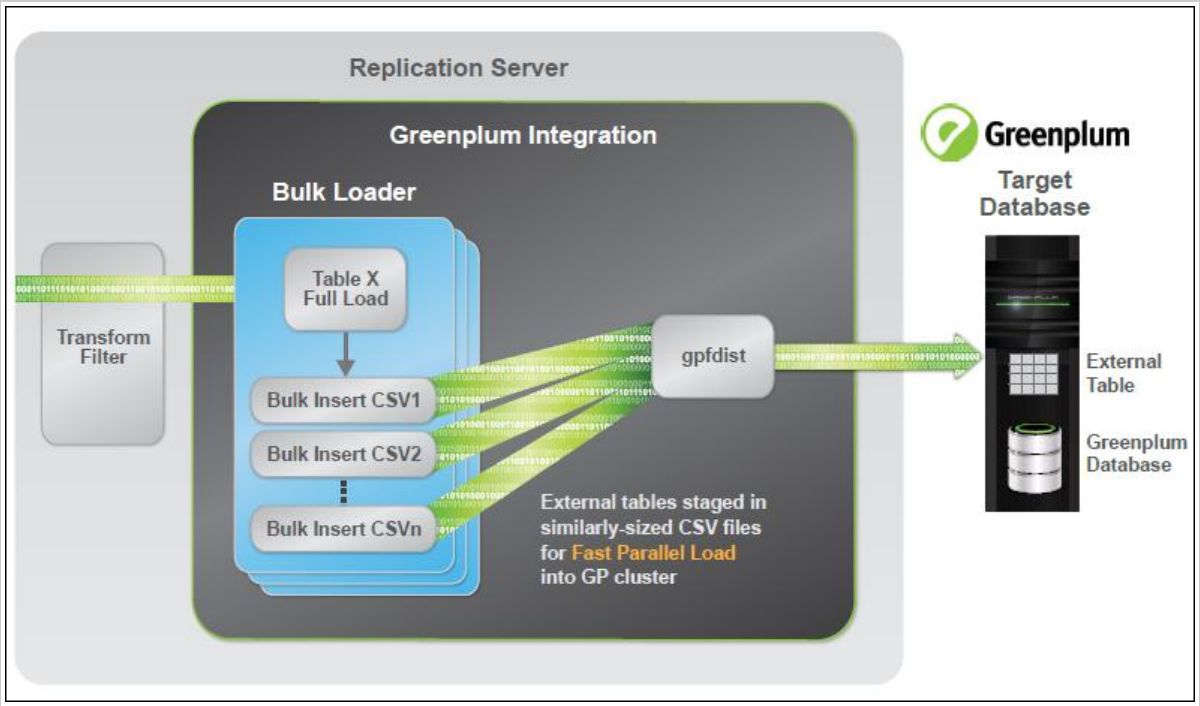
The following shows the Attunity Replicate Pivotal Greenplum database system architecture for:

- Full Load
- CDC

Full Load

Full load is used to setup or refresh a data warehouse on a target, by concurrently loading large amounts of data from source tables. High-speed data extraction is initiated from endpoints like Oracle or Microsoft SQL Server, then gpfdist and buffered load files are used for high-speed data loading into Pivotal Greenplum. The following shows the Pivotal Greenplum database architecture for full load.

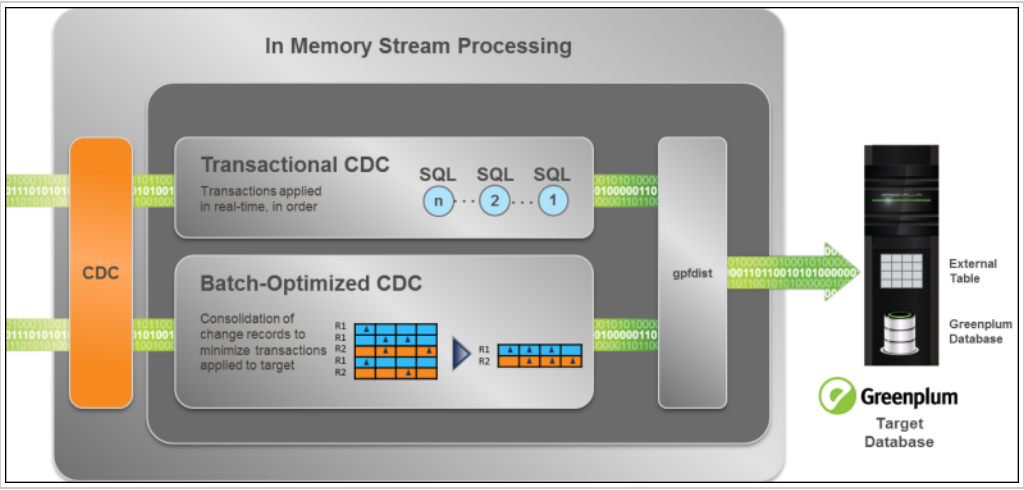
Figure 9.1 | Pivotal Greenplum database Full Load



CDC

For incremental load, Attunity Replicate uses log-based change data capture (CDC). During CDC replication, Attunity Replicate creates external Web tables or external tables to load SQL statements into the target Pivotal Greenplum database. The statements are then applied to the target tables. The following shows the Pivotal Greenplum database architecture for CDC.

Figure 9.2 | Pivotal Greenplum database CDC



Full Load

On the first run of a task the Pivotal Greenplum target writes the data being replicated to CSV files into a folder that is defined for the task. The CSV files are named sequentially, for example, loadNNNN, where NNNN is an incremental number starting from 0. The maximum file size of the CSV file is set by the user when configuring the Pivotal Greenplum database.

When the CSV file reaches its maximum size it is renamed and moved into a load folder. It is then read by the gpfdist utility, which executes an SQL statement that loads the data into the target table. Once the file loading is complete, the file is deleted.

Applying Changes to the Pivotal Greenplum Target

You can apply changes in one of two modes:

- Transactional Apply Mode
- Batch-Optimized Apply Mode

For information on how to use the transactional apply mode, see Apply Changes Settings.

Transactional Apply Mode

In this mode, the Pivotal Greenplum database writes all the change records to CSV files as DML statement. When a file is ready, the Pivotal Greenplum database creates an external Web table that uses the gpfdist server to read changes from the file and executes the DML statements in each row returned from the external Web table. When the changes are applied, the file is deleted.

Batch-Optimized Apply Mode

In this mode, the Pivotal Greenplum database writes net changes only to CSV files. When a file is ready, the Pivotal Greenplum database uses an external table that uses the gpfdist server to read the net changes from the file to a temporary table. The net changes are then applied to the target tables in the most efficient way. When the changes are applied, the file is deleted.

Required Pivotal Greenplum Software, Environments

You can use the Pivotal Greenplum database with Attunity Replicate on either a Windows or Linux computer. The following sections describe the prerequisites necessary to prepare your environment to work with Attunity Replicate and a Pivotal Greenplum database.

- Windows Pivotal Greenplum Required Software
- Linux Pivotal Greenplum Required Software
- Required Pivotal Greenplum Configuration and Environment

Windows Pivotal Greenplum Required Software

You must install the following on the same computer where the Attunity Replicate Server is installed:

- Pivotal Greenplum Client Tools 4.2.x
- Pivotal Greenplum Connectivity Tools 4.2.x
- Pivotal Greenplum Loaders 4.2.x (This will install the gpfdist program)
- Pivotal Greenplum DataDirect ODBC Drivers Version 7.x (only).

  You can download the ODBC drivers from emc.subscribenet.com.

Important: To prevent errors during replication tasks, make sure that the Pivotal Greenplum ODBC driver has a valid license.

Linux Pivotal Greenplum Required Software

You must make sure to configure the Linux computer as follows:

- Pivotal Greenplum Connectivity Tools 4.2.1.0. Unzip and install all of the components.
- Install and configure **UnixODBC**.
- Ensure that the following Pivotal Greenplum environment scripts are executed when logging into the account where Attunity Replicate is run:
  - greenplum_connectivity_path.sh
  - greenplum_clients_path.sh
  - greenplum_loaders_path.sh
- Make sure that port 8080 is open.

Required Pivotal Greenplum Configuration and Environment

Attunity Replicate relies on the proper functioning of the Pivotal Greenplum's gpfdist program on the computer with Attunity Replicate (the local computer). gpfdist is a simple Web server program with special performance customization for concurrent access from Pivotal Greenplum database segment nodes to data files on the local computer.

Because gpfdist is a Web server program and because it needs to be accessible from the Pivotal Greenplum database segment nodes, there are some networking configuration settings that must be in place to allow for this access. This is documented in the *EMC Pivotal Greenplum database Administration Guide*.

For further information, see Pivotal Greenplum Prerequisites for Attunity Replicate.

Provide Pivotal Greenplum Account Access

The Attunity Replicate user who is working with the Pivotal Greenplum database must be registered as a user in the Pivotal Greenplum database. This is the user that is entered in the dialog box when Setting General Connection Properties. You must grant Pivotal

Greenplum account access to this user before configuring the database in Attunity Replicate.

Note One of the Attunity Replicate computer network cards must be part of the Pivotal Greenplum database segments network to allow communication with the gpfdist program.

See Required Pivotal Greenplum Configuration and Environment for additional information about connecting to and configuring a Pivotal Greenplum database to work with Attunity Replicate.

Security Requirements

A user must have the following privileges granted in the Pivotal Greenplum database to use a Pivotal Greenplum target in an Attunity Replicate task:

- CREATE table/external table/Web external table
- TRUNCATE table
- ALTER table
- INSERT/UPDATE/DELETE

Limitations

Attunity Replicate cannot update columns that are part of the distribution key.

The Pivotal Greenplum target database has limited LOB support. You cannot use an unlimited LOB size for this database. For more information, see Pivotal Greenplum Data Types.

Pivotal Greenplum Data Types

The Pivotal Greenplum database for Attunity Replicate supports most Pivotal Greenplum data types. The following table shows the Pivotal Greenplum target data types that are supported when using Attunity Replicate and the default mapping from Attunity Replicate data types.

For information on how to view the data type that is mapped from the source, see the section for the source database you are using. For additional information about Attunity Replicate data types, see Replicate Data Types.

Table 9.13 | Supported Pivotal Greenplum Data Types with Mapping from Attunity Replicate Data Types

Attunity Replicate Data Types	Pivotal Greenplum Data Types
BOOLEAN	bool
BYTES	bytea
DATE	date
TIME	time (p)
DATETIME	timestamp (p)
INT1	int2
INT2	int2
INT4	int4
INT8	int8
NUMERIC	numeric (p,s)
REAL4	float4
REAL8	float8
STRING	varchar (n); n=data length
UINT1	int2
UINT2	int4
UINT4	int8
UINT8	numeric (20)
WSTRING	varchar (n); n=length
BLOB	bytea To use this data type with Attunity Replicate, you must enable the use of BLOBs for a specific task. LOB data types are supported only in tables that include a primary key. The Pivotal Greenplum target database has limited LOB support. You cannot use an unlimited LOB size for this database. For more information, see **Task Settings**/Metadata.
CLOB	text To use this data type with Attunity Replicate, you must enable the use of CLOBs for a specific task.

Table 9.13 | Supported Pivotal Greenplum Data Types with Mapping from Attunity Replicate Data Types (Cont.)

Attunity Replicate Data Types	Pivotal Greenplum Data Types
	LOB data types are supported only in tables that include a primary key. The Pivotal Greenplum target database has limited LOB support. You cannot use an unlimited LOB size for this database. For more information, see **Task Settings**/Metadata.
NCLOB	text To use this data type with Attunity Replicate, you must enable the use of NCLOBs for a specific task. LOB data types are supported only in tables that include a primary key. The Pivotal Greenplum target database has limited LOB support. You cannot use an unlimited LOB size for this database. For more information, see **Task Settings**/Metadata.

Setting up the gpfdist Program as a Service

You can use the gpfdist program as a service when replicating data to Pivotal Greenplum endpoints using Attunity Replicate. In this way it is assured that a replication task that is running will not lose its connection to gpfdist. This solution is best used when more than one task is using the same gpfdist program.

Note that you can set up Pivotal Greenplum endpoints as described in Setting General Connection Properties and test it without creating a service, but you should create the service before you begin to work with the Pivotal Greenplum endpoints.

Note If you want to use multiple gpfdist programs, then you should set them up on different ports. See Using Multiple gpfdist Programs.

To create use a gpfdist service:

1. From the command-line console on the computer where Attunity Replicate is installed, Change the directory to the directory where Attunity Replicate is installed.
2. Type the following at the command-line prompt to create the service.

   ```
   repctl service create <name of service (optional)> database=<name of Pivotal Greenplum database>
   ```

Note The Pivotal Greenplum database that you use in this command must be configured in Attunity Replicate. If you have defined more than one Pivotal Greenplum database, you can use any of the database names. All of the defined endpoints on the same Attunity Replicate instance are included in the defined service.

3. Start the service before you run any Pivotal Greenplum tasks by typing the following at the command line prompt:

   `sc start AttunityReplicateServer_<name of database you defined the service with>`

Note If you chose to create a data folder in a separate location from where you installed Attunity Replicate, you must add the `prefix -d <path to data folder>` before any command line task described in this section. For example, to start the service you must type:

`-d <path to data folder> sc start AttunityReplicateServer_<name of database>`

Working with a data folder in a different location is recommended when working with large Pivotal Greenplum endpoints. For more information, see Attunity Replicate on Windows: Installing, Upgrading and Uninstalling.

To stop the service:

» Type the following at the command-line prompt:

`sc stop AttunityReplicateServer_<name of database you defined the service with>`

To delete the service:

» Type the following at the command-line prompt:

`sc delete AttunityReplicateServer_<name of database you defined the service with>`

Note A log file is available in the Attunity Replicate data folder and in the Pivotal Greenplum debug files, which is accessed through the Pivotal Greenplum database console.

Using Multiple gpfdist Programs

You can define each replication task that has a Pivotal Greenplum database as a target to use a separate gpfdist program. In this case each gpfdist should run on a different port. If you want to use the same gpfdist program for each task, you may want to use the gpfdist program as a service. For more information, see Setting up the gpfdist Program as a Service.

To use gpfdist on multiple ports:

1. Install gpfdist programs on the computers you want to use them. See Pivotal Greenplum Prerequisites for Attunity Replicate.
2. Make sure that you assign a different port number for the gpfdist program in each task. For more information, see Setting General Connection Properties.

Setting General Connection Properties

This section describes how to configure general connection properties. For an explanation of how to configure advanced connection properties, see **Setting Advanced Connection Properties** below.

To add a Pivotal Greenplum Target endpoint to Attunity Replicate:

1. In **Tasks** view, click **Manage Endpoint Connections** to open the **Manage Endpoint Connections** dialog box.
2. In the **Manage database Connections** dialog box, click **New Endpoint Connection**.
3. In the **Name** field, type a name for your database. This can be any name that will help to identify the database being used.
4. In the **Description** field, type a description that helps to identify the Pivotal Greenplum database. This is optional.
5. Select **TARGET** as the database **role**.
6. Select **Pivotal Greenplum** as the database **Type**.
7. Type the **Database host name**. This is the name of the computer with the Pivotal Greenplum instance you want to work with.

   Note Although the gpfdist program acts as a Webserver, it does not carry out security checks on any requests made to it. Therefore, when you define the path to the gpfdist program, it must be to a specific location so that no other data on the computer is accessed.

   You can use the **Advanced** tab to add specific properties and create a custom connect string. In this case, you do not need to enter information in this tab. For more information on using the **Advanced** tab, see Setting Advanced Connection Properties.

8. Type the Pivotal Greenplum database Port, where the Pivotal Greenplum instance you are working with is located. The default value is 5432.
9. Type the Pivotal Greenplum authentication information (**User Name**, **Password**) for the authorized user for this Pivotal Greenplum database. If you do not know this information, see your Pivotal Greenplum system manager.

Note

- If you are using the **Advanced** tab to create a custom string, make sure to include the **User Name** property. A **Password** can also be included but is not required. See Setting Advanced Connection Properties for more information.
- This information is case sensitive.
- If you want to set custom properties for this database, see Setting Advanced Connection Properties.
- To determine if you are connected to the database you want to use or if the connection information you entered is correct, click **Test Connection**

  If the connection is successful a message in green is displayed. If the connection fails, an error message is displayed at the bottom of the dialog box.

  To view the log entry if the connection fails, click **View Log**. The server log is displayed with the information for the connection failure. Note that this button is not available unless the test connection fails.

Important: Make sure that the Pivotal Greenplum user entered in the Pivotal Greenplum Authentication section has the correct access privileges. For information on how to provide the required privileges, see Security Requirements.

10. Type the **Database name** or select one from the list of available databases. This is the name of the Pivotal Greenplum database where you are replicating the data to.
11. Type the **gpfdist hostname** for the server where the gpfdist program is installed.
12. Type the **gpfdist port** number where the gpfdist program is listening. The default value is 8080.

Setting Advanced Connection Properties

In the **Advanced** tab, you can set the following parameters:

- **gpfdist max row length**: Select or type the maximum length of a row (number of characters) in the CSV file that is sent to the gpfdist program. This is the maximum row length read by gpfdist. The default value is 32,768. The larger the size of the rows the more resources the gpfdist program uses.
- **Create tables in tablespace**: Type the name of the tablespace where you want create the target tables. This is optional. Note that the tablespace that you enter must exist in the Pivotal Greenplum database.
- **Max file size** (KB): Select or type the maximum size (in KB) of a CSV file before the file is moved into the load folder. The default value is 32000 KB.
- **Write buffer size**: (KB): Select or type the maximum amount of memory (in Kilobytes) used to store the data before moving it to the load folder. The default value is 1001.

- **ODBC driver**: Type the name of the ODBC driver you are using to connect to the Pivotal Greenplum database you are working with. The default value is **DataDirect 7.0 Pivotal Greenplum Wire Protocol**.
- **Additional ODBC connection properties**: Type any additional ODBC connection properties if required
- **Use externally managed gpfdist**: Select this check box to use an external gpfdist with this Pivotal Greenplum database.
- **Storage folder**: Type the name and location (enter full path) of the folder that holds the CSV files for loading. This is available only if you are using an externally managed gpfdist.

Internal Parameters

Internal parameters are parameters that are not exposed in the UI. You should only use them if instructed by Attunity Support.

To add internal Attunity Replicate parameters:

1. Click the **Internal Parameters** link.

   The **Internal Parameters** dialog box opens.
2. In the edit box, type the name of the parameter you need to add and then click it.
3. The parameter is added to the table below the search box with its default value.
4. Change the default value as required.
5. To reset the parameter value to its default, click the "Restore default value" icon at the end of the row.

Settings Summary

You can view a summary of your settings by clicking the **Setting Summary** link. This is useful if you need to send a summary of your settings to Attunity Support.

Using Pivotal HAWQ as a Target

This section describes how to set up and use a Pivotal HAWQ database as a target in a replication task.

In this section:

An Overview of the Pivotal HAWQ Target

The Attunity Replicate database for Pivotal HAWQ is a powerful operational data warehousing solution that manages Big Data analytics and challenges. Attunity Replicate uses Pivotal HAWQ's Scatter/Gather Streaming technology to help with data integration. This technology handles large amounts of data well.

The Attunity Replicate Pivotal HAWQ database makes it possible to load data from other heterogeneous data sources and maintain the most up to date information. This is done by capturing changes and streaming the changes to the Pivotal HAWQ data warehouse. This can be done with a very low impact on the source data.

The Attunity Replicate Pivotal HAWQ database provides full automation for:

- Schema generation and data type mapping
- Full load of source database tables
- Incremental load of changes made to source tables
- Application of schema changes (DDL) made to the source tables.
- Synchronization between full load and CDC processes.

Manual control is also available if needed.

The Attunity Replicate Pivotal HAWQ database integrates with the Pivotal HAWQ database in two ways:

- **Pivotal HAWQ ODBC API**. This is used for metadata management. The Pivotal HAWQ ODBC API lets Attunity Replicate test the database connection, get the table list and the table schema, build procedures that create external tables to process a file, and invoke the procedures that load the destination table or apply changes from the external table. During the schema generation, data types can be mapped, such as Pivotal HAWQ to Postgres. Primary keys and distribution clauses are generated based on the primary key.
- **Pivotal HAWQ Parallel File Distribution Server (gpfdist)**. This utility is used with read-only external tables for fast, parallel data loading into a Pivotal HAWQ data warehouse. gpfdist uses maximum parallelism while reading from external tables.

  Attunity Replicate works closely with gpfdist to take advantage of its optimized fast, parallel loading facilities. Attunity Replicate uses the Pivotal HAWQ Parallel File Distribution Server to support both full load and incremental load activities.

See Attunity Replicate Pivotal HAWQ database Architecture Overview for a description of the system architecture used with the Pivotal HAWQ database.

Attunity Replicate Pivotal HAWQ database Architecture Overview

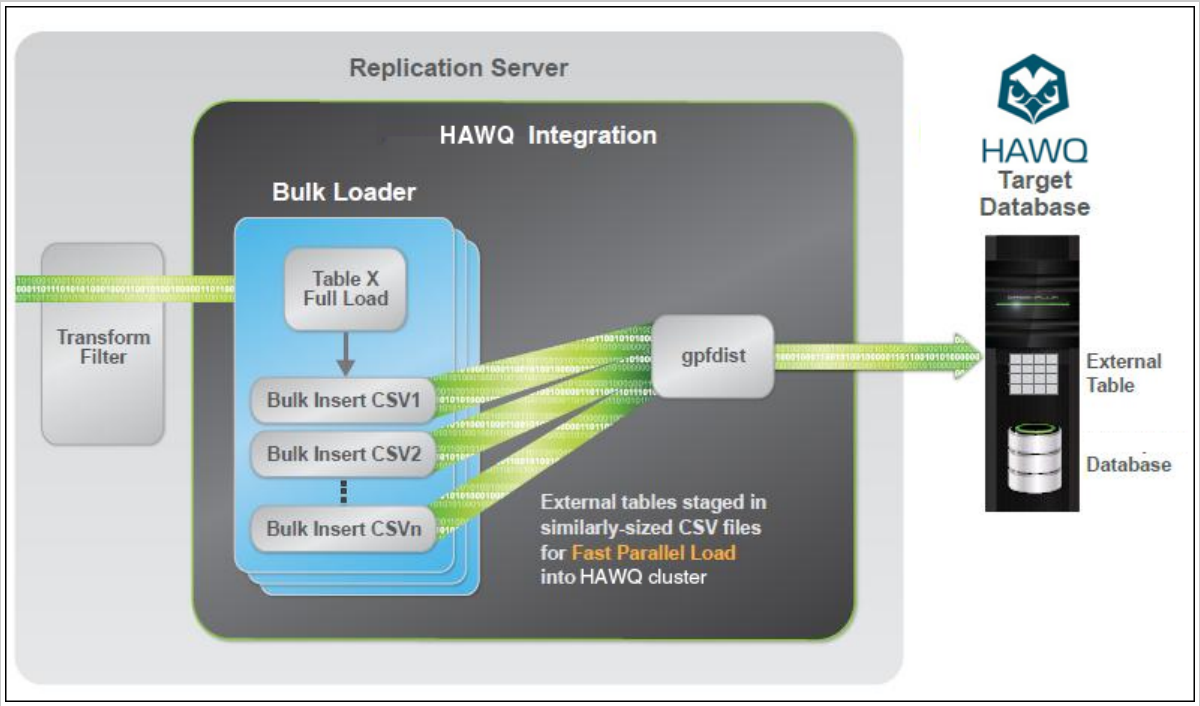
The following shows the Attunity Replicate Pivotal HAWQ database system architecture for:

- Full Load
- CDC

Full Load

Full load is used to setup or refresh a data warehouse on a target, by concurrently loading large amounts of data from source tables. High-speed data extraction is initiated from endpoints like Oracle or Microsoft SQL Server, then gpfdist and buffered load files are used for high-speed data loading into Pivotal HAWQ. The following shows the Pivotal HAWQ database architecture for full load.

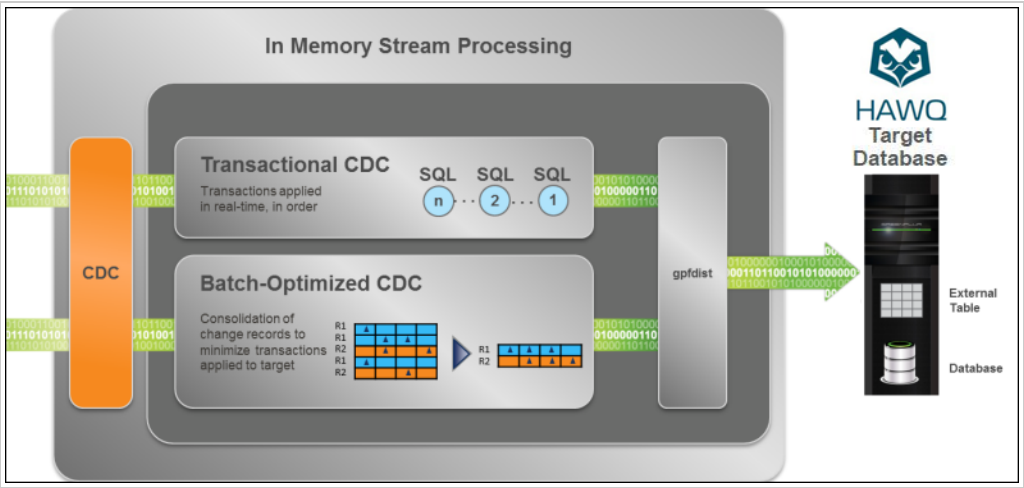
Figure 9.3 | Pivotal HAWQ database Full Load



CDC

For incremental load, Attunity Replicate uses log-based change data capture (CDC). During CDC replication, Attunity Replicate creates external Web tables or external tables to load SQL statements into the target Pivotal HAWQ database. The statements are then applied to the target tables. The following shows the Pivotal HAWQ database architecture for CDC.

Figure 9.4 | Pivotal HAWQ database CDC



Full Load

On the first run of a task the Pivotal HAWQ target writes the data being replicated to CSV files into a folder that is defined for the task. The CSV files are named sequentially, for example, loadNNNN, where NNNN is an incremental number starting from 0. The maximum file size of the CSV file is set by the user when configuring the Pivotal HAWQ database.

When the CSV file reaches its maximum size it is renamed and moved into a load folder. It is then read by the gpfdist utility, which executes an SQL statement that loads the data into the target table. Once the file loading is complete, the file is deleted.

Applying Changes to the Pivotal HAWQ Target

You can apply changes in one of two modes:

- Transactional Apply Mode
- Batch-Optimized Apply Mode

For information on how to use the transactional apply mode, see Apply Changes Settings.

Transactional Apply Mode

In this mode, the Pivotal HAWQ database writes all the change records to CSV files as DML statement. When a file is ready, the Pivotal HAWQ database creates an external Web table that uses the gpfdist server to read changes from the file and executes the DML statements in each row returned from the external Web table. When the changes are applied, the file is deleted.

Batch-Optimized Apply Mode

In this mode, the Pivotal HAWQ database writes net changes only to CSV files. When a file is ready, the Pivotal HAWQ database uses an external table that uses the gpfdist server to read the net changes from the file to a temporary table. The net changes are then applied to the target tables in the most efficient way. When the changes are applied, the file is deleted.

Required Pivotal HAWQ Software, Environments

You can use the Pivotal HAWQ database with Attunity Replicate on either a Windows or Linux computer. The following sections describe the prerequisites necessary to prepare your environment to work with Attunity Replicate and a Pivotal HAWQ database.

- Windows Pivotal HAWQ Required Software
- Linux Pivotal HAWQ Required Software
- Required Pivotal HAWQ Configuration and Environment

Windows Pivotal HAWQ Required Software

You must install the following on the same computer where the Attunity Replicate Server is installed:

- Pivotal HAWQ Client Tools 4.2.x
- Pivotal HAWQ Connectivity Tools 4.2.x
- Pivotal HAWQ Loaders 4.2.x (This will install the gpfdist program)
- Pivotal HAWQ DataDirect ODBC Drivers Version 7.x (only).

  You can download the ODBC drivers from emc.subscribenet.com.

> **Important:** To prevent errors during replication tasks, make sure that the Pivotal HAWQ ODBC driver has a valid license.

Linux Pivotal HAWQ Required Software

You must make sure to configure the Linux computer as follows:

- Pivotal HAWQ Connectivity Tools 4.2.1.0. Unzip and install all of the components.
- Install and configure **UnixODBC**.
- Ensure that the following Pivotal HAWQ environment scripts are executed when logging into the account where Attunity Replicate is run:
  - Pivotal HAWQ_connectivity_path.sh
  - Pivotal HAWQ_clients_path.sh
  - Pivotal HAWQ-loaders_path.sh
- Make sure that port 8080 is open.

Required Pivotal HAWQ Configuration and Environment

Attunity Replicate relies on the proper functioning of the Pivotal HAWQ's gpfdist program on the computer with Attunity Replicate (the local computer). gpfdist is a simple Web server program with special performance customization for concurrent access from Pivotal HAWQ database segment nodes to data files on the local computer.

Because gpfdist is a Web server program and because it needs to be accessible from the Pivotal HAWQ database segment nodes, there are some networking configuration settings that must be in place to allow for this access. This is documented in the *EMC Pivotal HAWQ database Administration Guide*.

For further information, see Pivotal Greenplum Prerequisites for Attunity Replicate.

Provide Pivotal HAWQ Account Access

The Attunity Replicate user who is working with the Pivotal HAWQ database must be registered as a user in the Pivotal HAWQ database. This is the user that is entered in the dialog box when Setting General Connection Properties. You must grant Pivotal HAWQ account access to this user before configuring the database in Attunity Replicate.

Note One of the Attunity Replicate computer network cards must be part of the Pivotal HAWQ database segments network to allow communication with the gpfdist program.

See Required Pivotal HAWQ Configuration and Environment for additional information about connecting to and configuring a Pivotal HAWQ database to work with Attunity Replicate.

Security Requirements

A user must have the following privileges granted in the Pivotal HAWQ database to use a Pivotal HAWQ target in an Attunity Replicate task:

- CREATE table/external table/Web external table
- TRUNCATE table
- ALTER table
- INSERT/UPDATE/DELETE

Limitations

The following limitations apply:

- Attunity Replicate cannot update columns that are part of the distribution key.
- The Pivotal HAWQ target database has limited LOB support. You cannot use an unlimited LOB size for this database. For more information, see Pivotal HAWQ Data Types.
- UPDATE and DELETE change operations are not supported. However, records of UPDATE and DELETE operations will still appear in the Change Table created on the target (providing the task's "Store changes in Changes tables" option is enabled).

  A warning will be issued if data has been updated in/deleted from the source tables.
- The "Status" and "Suspend" Control Tables are not supported.
- For more information on Control Tables, see Control Tables.
- The ALTER TABLE statement is not supported.

Pivotal HAWQ Data Types

The Pivotal HAWQ database for Attunity Replicate supports most Pivotal HAWQ data types. The following table shows the Pivotal HAWQ target data types that are supported when using Attunity Replicate and the default mapping from Attunity Replicate data types.

For information on how to view the data type that is mapped from the source, see the section for the source database you are using. For additional information about Attunity Replicate data types, see Replicate Data Types.

Table 9.14 | Supported Pivotal HAWQ Data Types with Mapping from Attunity Replicate Data Types

Attunity Replicate Data Types	Pivotal HAWQ Data Types
BOOLEAN	bool
BYTES	bytea
DATE	date
TIME	time (p)
DATETIME	timestamp (p)
INT1	int2
INT2	int2
INT4	int4
INT8	int8
NUMERIC	numeric (p,s)
REAL4	float4
REAL8	float8
STRING	varchar (n); n=data length
UINT1	int2
UINT2	int4
UINT4	int8
UINT8	numeric (20)
WSTRING	varchar (n); n=length
BLOB	bytea To use this data type with Attunity Replicate, you must enable the use of BLOBs for a specific task. LOB data types are supported only in tables that include a primary key. The Pivotal HAWQ target database has limited LOB support. You cannot use an unlimited LOB size for this database. For more information, see **Task Settings**/Metadata.
CLOB	text To use this data type with Attunity Replicate, you must enable the use of CLOBs for a specific task.

Table 9.14 | Supported Pivotal HAWQ Data Types with Mapping from Attunity Replicate Data Types (Cont.)

Attunity Replicate Data Types	Pivotal HAWQ Data Types
	LOB data types are supported only in tables that include a primary key. The Pivotal HAWQ target database has limited LOB support. You cannot use an unlimited LOB size for this database. For more information, see **Task Settings**/Metadata.
NCLOB	text To use this data type with Attunity Replicate, you must enable the use of NCLOBs for a specific task. LOB data types are supported only in tables that include a primary key. The Pivotal HAWQ target database has limited LOB support. You cannot use an unlimited LOB size for this database. For more information, see **Task Settings**/Metadata.

Setting up the gpfdist Program as a Service

You can use the gpfdist program as a service when replicating data to Pivotal HAWQ endpoints using Attunity Replicate. In this way it is assured that a replication task that is running will not lose its connection to gpfdist. This solution is best used when more than one task is using the same gpfdist program.

Note that you can set up Pivotal HAWQ endpoints as described in Setting General Connection Properties and test it without creating a service, but you should create the service before you begin to work with the Pivotal HAWQ endpoints.

Note If you want to use multiple gpfdist programs, then you should set them up on different ports. See Using Multiple gpfdist Programs.

To create use a gpfdist service:

1. From the command-line console on the computer where Attunity Replicate is installed, Change the directory to the directory where Attunity Replicate is installed.
2. Type the following at the command-line prompt to create the service.

   ```
   repctl service create <name of service (optional)> database=<name of Pivotal HAWQ database>
   ```

Note The Pivotal HAWQ database that you use in this command must be configured in Attunity Replicate. If you have defined more than one Pivotal HAWQ database, you can use any of the database names. All of the defined endpoints on the same Attunity Replicate instance are included in the defined service.

3. Start the service before you run any Pivotal HAWQ tasks by typing the following at the command line prompt:

   `sc start AttunityReplicateServer_<name of database you defined the service with>`

Note If you provided a name when you created the service, enter the service name instead of `AttunityReplicateServer`.

Note If you chose to create a data folder in a separate location from where you installed Attunity Replicate, you must add the `prefix -d <path to data folder>` before any command line task described in this section. For example, to start the service you must type:

`-d <path to data folder> sc start AttunityReplicateServer_<name of database>`

Working with a data folder in a different location is recommended when working with large Pivotal HAWQ endpoints. For more information, see Attunity Replicate on Windows: Installing, Upgrading and Uninstalling.

To stop the service:

- Type the following at the command-line prompt:

  `sc stop AttunityReplicateServer_<name of database you defined the service with>`

To delete the service:

- Type the following at the command-line prompt:

  `sc delete AttunityReplicateServer_<name of database you defined the service with>`

Note A log file is available in the Attunity Replicate data folder and in the Pivotal HAWQ debug files, which is accessed through the Pivotal HAWQ database console.

Using Multiple gpfdist Programs

You can define each replication task that has a Pivotal HAWQ database as a target to use a separate gpfdist program. In this case each gpfdist should run on a different port. If you want to use the same gpfdist program for each task, you may want to use the gpfdist

program as a service. For more information, see Setting up the gpfdist Program as a Service.

To use gpfdist on multiple ports:

1. Install gpfdist programs on the computers you want to use them. See Pivotal Greenplum Prerequisites for Attunity Replicate.
2. Make sure that you assign a different port number for the gpfdist program in each task. For more information, see Setting General Connection Properties.

Setting General Connection Properties

This section describes how to configure general connection properties. For an explanation of how to configure advanced connection properties, see **Setting Advanced Connection Properties** below.

For more information on how to add an endpoint to Attunity Replicate, see Working with Endpoints.

To add a Pivotal HAWQ target endpoint to Attunity Replicate:

1. In the Attunity Replicate console, click **Manage Endpoint Connections** to open the **Manage database Connections** dialog box.
2. In the **Manage Endpoint Connections** dialog box, click **New Endpoint Connection**.
3. In the **Name** field, type a name for your database. This can be any name that will help to identify the database being used.
4. In the **Description** field, type a description that helps to identify the Pivotal HAWQ database. This is optional.
5. Select **TARGET** as the database **role**.
6. Select **Pivotal HAWQ** as the database **Type**.

   Settings relevant to Pivotal HAWQ are displayed.
7. Type the **Database host name**. This is the name of the computer with the Pivotal HAWQ instance you want to work with.

   Note Although the gpfdist program acts as a Webserver, it does not carry out security checks on any requests made to it. Therefore, when you define the path to the gpfdist program, it must be to a specific location so that no other data on the computer is accessed.

   You can use the **Advanced** tab to add specific properties and create a custom connect string. In this case, you do not need to enter information in this tab. For more information on using the **Advanced** tab, see Setting Advanced Connection Properties.
8. Type the Pivotal HAWQ database Port, where the Pivotal HAWQ instance you are working with is located. The default value is 5432.

9. Type the Pivotal HAWQ authentication information (**User Name**, **Password**) for the authorized user for this Pivotal HAWQ database. If you do not know this information, see your Pivotal HAWQ system manager.

   Note Consider the following:

   - If you are using the **Advanced** tab to create a custom string, make sure to include the **User Name** property. A **Password** can also be included but is not required. See Setting Advanced Connection Properties for more information.
   - This information is case sensitive.
   - If you want to set custom properties for this database, see Setting Advanced Connection Properties.
   - To determine if you are connected to the database you want to use or if the connection information you entered is correct, click **Test Connection**.

     If the connection is successful a message in green is displayed. If the connection fails, an error message is displayed at the bottom of the dialog box.

     To view the log entry if the connection fails, click **View Log**. The server log is displayed with the information for the connection failure. Note that this button is not available unless the test connection fails.

   Important: Make sure that the Pivotal HAWQ user entered in the Pivotal HAWQ Authentication section has the correct access privileges. For information on how to provide the required privileges, see Security Requirements.

10. Type the **database name** or select one from the list of available databases. This is the name of the Pivotal HAWQ database where you are replicating the data to.
11. Type the **gpfdist hostname** for the server where the gpfdist program is installed.
12. Type the **gpfdist port** number where the gpfdist program is listening. The default value is 8080.

Setting Advanced Connection Properties

In the **Advanced** tab, you can set the following parameters:

- **gpfdist max row length**: Select or type the maximum length of a row (number of characters) in the CSV file that is sent to the gpfdist program. This is the maximum row length read by gpfdist. The default value is 32,768. The larger the size of the rows the more resources the gpfdist program uses.
- **Create tables in tablespace**: Type the name of the tablespace where you want create the target tables. This is optional. Note that the tablespace that you enter must exist in the Pivotal HAWQ database.
- **Max file size** (KB): Select or type the maximum size (in KB) of a CSV file before the file is moved into the load folder. The default value is 32000 KB.

- **Write buffer size**: (KB): Select or type the maximum amount of memory (in Kilo-bytes) used to store the data before moving it to the load folder. The default value is 1001.
- **ODBC driver**: Type the name of the ODBC driver you are using to connect to the Pivotal HAWQ database you are working with. The default value is **DataDirect 7.0 Pivotal HAWQ Wire Protocol**.
- **Additional ODBC connection properties**: Type any additional ODBC connection properties if required
- **Use externally managed gpfdist**: Select this check box to use an external gpfdist with this Pivotal HAWQ database.
- **Storage folder**: Type the name and location (enter full path) of the folder that holds the CSV files for loading. This is available only if you are using an externally managed gpfdist.

Internal Parameters

Internal parameters are parameters that are not exposed in the UI. You should only use them if instructed by Attunity Support.

To add internal Attunity Replicate parameters:

1. Click the **Internal Parameters** link.

   The **Internal Parameters** dialog box opens.
2. In the edit box, type the name of the parameter you need to add and then click it.
3. The parameter is added to the table below the search box with its default value.
4. Change the default value as required.
5. To reset the parameter value to its default, click the "Restore default value" icon at the end of the row.

Settings Summary

You can view a summary of your settings by clicking the **Setting Summary** link. This is useful if you need to send a summary of your settings to Attunity Support.

Using Actian Vector as a Target

This section describes how to set up and use an Actian Vector database as a target in a replication task.

In this section:

Prerequisites

An Actian Vector database with the tables that are being used for replication must be available to the system. Attunity Replicate supports using an Actian Vector database on both Windows and Linux.

Actian Vector 3.0 must be installed on the same machine as Attunity Replicate Server. From Actian Vector 3.5, with the addition of remote load support, Replicate and Actian Vector do not need to be installed on the same machine. When Actian Vector is installed on a remote machine (i.e. *not* on the Replicate machine), Client Runtime 3.5.1 or above for Linux/Windows needs to be installed on the Replicate machine.

For more information about the requirements for working with Attunity Replicate, see Installation Prerequisites.

The following sections describe the prerequisites necessary to prepare your environment to work with Attunity Replicate and an Actian Vector database:

- Actian Vector Windows Environment Prerequisites
- Actian Vector Linux Environment Prerequisites

Actian Vector Windows Environment Prerequisites

The following must be installed:

- Actian Vector database on any computer in your network. Make sure to configure the tables that you need for replication.
- Actian Vector DBA Tools. This includes the Ingres ODBC driver 3.0 or above. You must install the DBA tools on the same computer as the Actian Vector database.

  For more information, see the Actian Web site.
- vwload utility. Install this utility on the same computer as the Actian Vector database.

Actian Vector Linux Environment Prerequisites

You must install the following:

- Be sure to configure the Ingres ODBC driver. This driver supports unixODBC and is included in the Actian Vector database installation. unixODBC is built in to most Linux distributions.

  Linux releases of Ingres include the ODBC CLI. The CLI is an ODBC manager/library that is built specifically for Ingres. The CLI is installed on Linux platforms at the same time as the ODBC driver. The CLI allows ODBC applications to be developed and deployed without the need to manage or use unixODBC. Use the iiodbcadmin tool to manage DSN definitions (or use a connection string that specifies the database name). IIodbcadmin can be used with either unixODBC or CLI applications. For information on configuring the ODBC driver, go to the following page on the Actian Web site: http://community.actian.com/wiki/Ingres_UNIX_ODBC.

Limitations

The following DDL operations cannot be made to the Actian Vector database using Attunity Replicate:

- Change schema name
- Change column type

The replication task fails when applying DDL changes that were initiated in source but not supported on target.

Provide Actian Vector Account Access

The Attunity Replicate user who is working with the Actian Vector database must be registered as a user in the Actian Vector database. This is the user that is entered in the dialog box when Setting General Connection Properties. You must grant Actian Vector account access to this user before confiding the database in Attunity Replicate.

Actian Vector Data Types

The Actian Vector database for Attunity Replicate supports most Actian Vector data types. The following table shows the Actian Vector target data types that are supported when using Attunity Replicate and the default mapping from Attunity Replicate data types.

Note Actian Vector does not support applying changes to binary data types in **Batch optimized apply** mode. For more information on **Batch optimized apply** mode, see Changes Processing Tuning.

For information on how to view the data type that is mapped from the source, see the section for the source database you are using. For additional information about Attunity Replicate data types, see Replicate Data Types.

Table 9.15 | Supported Actian Vector Data Types with Mapping from Attunity Replicate Data Types

Attunity Replicate Data Types	Actian Vector Data Types
BOOLEAN	VARCHAR (length)
BYTES	If length is => 1 and =< 16000, then: VARCHAR (Length in Bytes) If length is => 16001 and =< 2147483647, then: VARCHAR (32000)
DATE	ANSIDATE
TIME	TIME
DATETIME	TIMESTAMP
INT1	INTEGER1
INT2	SMALLINT
INT4	INTEGER
INT8	INTEGER8
NUMERIC	DECIMAL (p,s)
REAL4	FLOAT4
REAL8	FLOAT
STRING	If length is => 1 and =< 32000, then: VARCHAR (Length) If length is => 32001 and =< 2147483647, then: VARCHAR (32000)
UINT1	SMALLINT
UINT2	INTEGER
UINT4	INTEGER8
UINT8	INTEGER8
WSTR	If length is => 1 and =< 16000, then: NVARCHAR (Length) If length is => 16001 and =< 2147483647, then: NVARCHAR (16000)

Note About Actian Vector LOB support

Table 9.15 | Supported Actian Vector Data Types with Mapping from Attunity Replicate Data Types (Cont.)

Attunity Replicate Data Types	Actian Vector Data Types
Full LOB data types are not supported. For information on including Limited-size LOB data types in the replication, see the Metadata tab description in Customizing Tasks	
BLOB	VARCHAR (16000) **Note** The maximum LOB size in the Metadata tab cannot exceed 15 KB.
NCLOB	NVARCHAR (16000) **Note** The maximum LOB size in the Metadata tab cannot exceed 15 KB.
CLOB	In **Batch optimized apply** mode: VARCHAR (32000) **Note** In **Batch optimized apply** mode, the maximum LOB size in the Metadata tab cannot exceed 31 KB. In **Transactional apply** mode: VARCHAR (16000) **Note** In **Transactional apply** mode, the maximum LOB size in the Metadata tab cannot exceed 31 KB. For more information on **Transactional apply** and **Batch optimized apply**, see Changes Processing Tuning.

Setting General Connection Properties

This section describes how to configure general connection properties. For an explanation of how to configure advanced connection properties, see **Setting Advanced Connection Properties** below.

To add an Actian Vector target endpoint to Attunity Replicate:

1. In **Tasks** view, click **Manage Endpoint Connections** to open the **Manage Endpoints Connections** dialog box. Then click the **New Endpoint Connection** button.
2. In the **Name** field, type a name for your database. This can be any name that will help to identify the database being used.
3. In the **Description** field, type a description that helps to identify the Greenplum database. This is optional.
4. Select **TARGET** as the database **role**.
5. Select **Actian Vector Target** as the database **Type**.
6. In the **Server** field, enter the hostname or IP address of the machine on which the Actian Vector database is installed.

   Note You can use the **Advanced** tab to add specific properties and create a custom connect string. In this case, you do not need to enter information in this tab. For more information on using the **Advanced** tab, see Setting Advanced Connection Properties.

7. Type the Actian Vector authentication information (**User name**, **Password**) for the authorized user for this Actian Vector database. If you do not know this information, see your Actian Vector system manager.

   Note

   - If you are using the **Advanced** tab to create a custom string, make sure to include the **User Name** property. A **Password** can also be included but is not required. See Setting Advanced Connection Properties for more information.
   - This information is case sensitive.
   - If you want to set custom properties for this database, see Setting Advanced Connection Properties.
   - To determine if you are connected to the database you want to use or if the connection information you entered is correct, click **Test Connection**.

     If the connection is successful a message in green is displayed. If the connection fails, an error message is displayed at the bottom of the dialog box.

     To view the log entry if the connection fails, click **View Log**. The server log is displayed with the information for the connection failure. Note that this button is not available unless the test connection fails.

Setting Advanced Connection Properties

In the **Advanced** tab, you can set the following parameters:

- **Max file size** (KB): Select or type the maximum file size. When the Actian Vector file reaches this value, the data is loaded by the vmload utility. The default value is 300000 KB.

Internal Parameters

Internal parameters are parameters that are not exposed in the UI. You should only use them if instructed by Attunity Support.

To add internal Attunity Replicate parameters:

1. Click the **Internal Parameters** link.

   The **Internal Parameters** dialog box opens.
2. In the edit box, type the name of the parameter you need to add and then click it.
3. The parameter is added to the table below the search box with its default value.
4. Change the default value as required.
5. To reset the parameter value to its default, click the "Restore default value" icon at the end of the row.

Settings Summary

You can view a summary of your settings by clicking the **Setting Summary** link. This is useful if you need to send a summary of your settings to Attunity Support.

Using Amazon Redshift as a Target

This section describes how to set up and use Amazon Redshift as a target in a replication task. Amazon Redshift is located in the cloud and is accessed through an Amazon Web Services (AWS) account.

In this section:

Introducing the Amazon Redshift Target Endpoint for Attunity Replicate

Amazon Redshift is a fully-managed petabyte-scale data warehouse service in the cloud.

In the first stage of the replication process, Attunity Replicate moves the data files created by the source database into an Amazon S3 bucket. The files are then loaded into the proper tables in the Amazon Redshift data warehouse (using the "copy" command).

The Amazon Redshift database provides full automation for:

- Schema generation and data type mapping
- Full load of source database tables
- Incremental load of changes made to source tables
- Application of schema changes (DDL) made to the source tables.
- Synchronization between full load and CDC processes.

Manual control is also available if needed.

Limitations

The following limitations apply to the Amazon Target endpoint:

- The `ALTER TABLE <NAME> MODIFY COLUMN <NAME> <DATA_TYPE> DDL` is not supported.
- Applying a DELETE statement to a table with a multi-column primary key is not supported if any of the primary key column names is a reserved word (e.g. "tag").

  For a list of reserved Amazon Redshift words, see http://docs.aws.amazon.com/redshift/latest/dg/r_pg_keywords.html

Amazon Redshift Database Prerequisites

The following sections describe the prerequisites necessary for working with the Amazon Redshift database:

- Get Started with Amazon Redshift
- Sign up for an Amazon S3 Bucket
- Open the Required Firewall Port

Get Started with Amazon Redshift

Once you register for an Amazon Web Services (AWS) account, you can launch an Amazon Redshift cluster and download the required SQL client tools. The following describes what you need to do to get started using Amazon Redshift as an Attunity Replicate target database.

- Sign up for an Amazon Web Services account. Then use the AWS Management Console to launch an Amazon Redshift cluster. You should note the basic information about your AWS account and your Amazon Redshift cluster, such as your password and user name. You will need this information to configure Attunity Replicate to work with the Amazon Redshift database. For more information, see Setting General Connection Parameters.
- Download and install the Windows or Linux SQL client tools (according to your Replicate Server platform) necessary to connect to the Amazon Redshift cluster. Attunity Replicate requires that you download a 64-bit ODBC driver.

  For a list of drivers supported by Amazon Redshift, see http://docs.aws.amazon.com/redshift/latest/mgmt/configure-odbc-connection.html.

  By default, Attunity Replicate uses the Amazon Redshift (x64) driver. If you use a different driver, you must change this in the Amazon Redshift database settings in the Attunity Replicate Console. For more information, see Using Advanced Properties for an Amazon Redshift Target.

  Note To avoid conflicts when installing the driver on a Linux machine, Attunity Replicate must be installed *before* you install the driver. Install the Amazon Redshift ODBC driver with `--force` in the command, as in the following example:

  ```
  rpm -ivh AmazonRedshiftODBC-64bit-1.2.6.1006-1.x86_64.rpm --force
  ```

  Once the driver is installed, edit the **amazon.redshiftodbc.ini** file as follows:

```
DriverManagerEncoding=UTF-16
ODBCInstLib=libodbcinst.so
```

For information on signing up for an Amazon Web Services account, launching an Amazon Redshift cluster, and installing the client tools, see the Amazon Redshift Getting Started page at http://docs.aws.amazon.com.

Sign up for an Amazon S3 Bucket

You need to have an Amazon S3 bucket, preferably (for best performance) located in your Snowflake cluster region.You must be able to access your Amazon S3 bucket directly from the machine.

For information on signing up for Amazon S3, see http://aws.amazon.com/s3/.

- Bucket access credentials: Make a note of the bucket name, region, access key and secret access key - you will need to provide them in the Snowflake target endpoint settings.
- Bucket access permissions: requires read/write/delete permissions to the Amazon S3 bucket.

Open the Required Firewall Port

Firewall port 5439 (Amazon Redshift Cluster) need to be opened for outbound communication.

Amazon Redshift Data Types

The Amazon Redshift database for Attunity Replicate supports most Amazon Redshift data types. The following table shows the Amazon Redshift target data types that are supported when using Attunity Replicate and the default mapping from Attunity Replicate data types.

Note Amazon Redshift does not support applying changes to binary data types in **Batch optimized apply** mode. For more information on **Batch optimized apply** mode, see Changes Processing Tuning.

For information on how to view the data type that is mapped from the source, see the section for the source database you are using. For additional information about Attunity Replicate data types, see Replicate Data Types.

Table 9.16 | Supported Amazon Redshift Data Types with Mapping from Attunity Replicate Data Types

Attunity Replicate Data Types	Amazon Redshift Data Types
BOOLEAN	BOOL

Table 9.16 | Supported Amazon Redshift Data Types with Mapping from Attunity Replicate Data Types (Cont.)

Attunity Replicate Data Types	Amazon Redshift Data Types
BYTES	VARCHAR (Length)
DATE	DATE
TIME	VARCHAR(20)
DATETIME	If scale is => 0 and =< 6, then: TIMESTAMP (s) If scale is => 7 and =< 9, then: VARCHAR (37)
INT1	INT2
INT2	INT2
INT4	INT4
INT8	INT8
NUMERIC	If scale is => 0 and =< 37, then: NUMERIC (p,s) If scale is => 38 and =< 127, then: VARCHAR (Length)
REAL4	FLOAT4
REAL8	FLOAT8
STRING	VARCHAR (Length multiplied by three) For example, STRING (50) becomes VARCHAR (150).
UINT1	INT2
UINT2	INT2
UINT4	INT4
UINT8	NUMERIC (20,0)
WSTRING	If length is => 1 and =< 65535, then: NVARCHAR (Length in Bytes) If length is => 65536 and =< 2147483647, then: NVARCHAR (65535)

Note About Amazon Redshift LOB support:

Table 9.16 | Supported Amazon Redshift Data Types with Mapping from Attunity Replicate Data Types (Cont.)

Attunity Replicate Data Types	Amazon Redshift Data Types
Full LOB data types are not supported. For information on including Limited-size LOB data types in the replication, see the Metadata tab description in Customizing Tasks .	
BLOB	VARCHAR (Max LOB Size *2) **Note** The maximum LOB size in the Metadata tab cannot exceed 31 KB.
NCLOB	NVARCHAR (Max LOB Size) **Note** The maximum LOB size in the Metadata tab cannot exceed 63 KB.
CLOB	VARCHAR (Max LOB Size) **Note** The maximum LOB size in the Metadata tab cannot exceed 63 KB.

Setting General Connection Parameters

This section describes how to configure general connection properties. For an explanation of how to configure advanced connection properties, see **Setting Advanced Connection Properties** below.

To add an Amazon Redshift Target to Attunity Replicate:

1. In the Attunity Replicate Console, click Manage Endpoint Connections to open the **Manage Endpoints Connections** dialog box.
2. In the **Manage Endpoint Connections** dialog box, click **New Endpoint Connection**.
3. In the **Name** field, type a name for your Amazon Redshift data warehouse [service]. This can be any name that will help to identify your Amazon Redshift database.
4. Optionally, in the **Description** field, type a description that helps to identify the Amazon Redshift target database.
5. Select **TARGET** as the **role**.
6. Select **Amazon Redshift** as the **Type**.
7. Enter the following **Amazon Redshift target** information:
   - **Redshift cluster**: Type the name of the Amazon Redshift cluster you are using.
   - **Port**: Type the port number for Amazon Redshift.

- **User name**: Type an Amazon Redshift user name for a registered user.
- **Password**: Type the password for the user entered in the **User name** field.
- **Database name**: Type the **database name** or select one from the list of available Amazon Redshift data warehouse [services].

The information for these properties is available from the account page for Amazon Web Services (AWS) with the Amazon Redshift cluster. If you do not have these values, refer to your AWS account or the Amazon Redshift System Administrator for your enterprise.

8. Enter the following **Amazon S3 staging** information. You may need to click the **Amazon S3 staging** header to see the information.
   - **Bucket name**: Type the name of the Amazon S3 bucket where you are copying files to.
   - **Bucket region**: Select the Amazon S3 region where the S3 buckets and folders you are using are hosted. The default value is **US East (N. Virginia)**.

     Note: The bucket region specified must be the same region where your Amazon Redshift database is located.
   - **Access key**: Type the access key information for Amazon S3.
   - **Secret key**: Type the secret key information for Amazon S3.
   - **Folder**: Type or browse to the S3 folder where you are copying files to.

   The information for these properties is available from your Amazon Web Services (AWS) account. If you do not have these values, refer to your AWS account or the Amazon Redshift System Administrator for your enterprise

Note

- This information is case sensitive.
- To determine if you are connected to the database you want to use or if the connection information you entered is correct, click **Test Connection**.

  If the connection is successful a message in green is displayed. If the connection fails, an error message is displayed at the bottom of the dialog box.

  To view the log entry if the connection fails, click **View Log**. The server log is displayed with the information for the connection failure. Note that this button is not available unless the test connection fails.

Using Advanced Properties for an Amazon Redshift Target

In the **Advanced** tab, you can set the following parameters:

- **Max file size (MB)**: Select or type the maximum size of any CSV file used to transfer data to Amazon Redshift. The default value is 1024.
- **Number of threads used to upload a file**: Select the number of threads used to upload a single file. The minimum number of threads is 1. The maximum value is 64. The default value is 10.

- **ODBC driver**: The name of the default ODBC driver you are using to connect to Amazon Redshift. The default value is **Amazon Redshift (x64)**.
- **Additional ODBC connection properties**: Type any additional ODBC connection properties if required.

Internal Parameters

Internal parameters are parameters that are not exposed in the UI. You should only use them if instructed by Attunity Support.

To add internal Attunity Replicate parameters:

1. Click the **Internal Parameters** link.
   The **Internal Parameters** dialog box opens.
2. In the edit box, type the name of the parameter you need to add and then click it.
3. The parameter is added to the table below the search box with its default value.
4. Change the default value as required.
5. To reset the parameter value to its default, click the "Restore default value" icon at the end of the row.

Settings Summary

You can view a summary of your settings by clicking the **Setting Summary** link. This is useful if you need to send a summary of your settings to Attunity Support.

Using Snowflake as a Target

This section describes how to set up and use Snowflake as a target in a replication task. Snowflake is located in the cloud and is accessed through an Amazon Web Services (AWS) account.

In this section:

Introducing the Snowflake Target Endpoint for Attunity Replicate

Snowflake is a fully-managed petabyte-scale data warehouse service in the cloud.

In the first stage of the replication process, Attunity Replicate moves the data files created by the source database into an Amazon S3 bucket. The files are then loaded into the proper tables in the Snowflake data warehouse (using the "copy" command).

The Snowflake database provides full automation for:

- Schema generation and data type mapping
- Full load of source database tables
- Incremental load of changes made to source tables
- Application of schema changes (DDL) made to the source tables.
- Synchronization between full load and CDC processes.

Manual control is also available if needed.

Snowflake Database Prerequisites

The following section describes the prerequisites necessary for working with the Snowflake database:

- Get Started with Snowflake
- Sign up for an Amazon S3 Bucket
- Open the Required Firewall Port

Get Started with Snowflake

Once you register for an Amazon Web Services (AWS) account, you can launch an Snowflake cluster and download the required client tools. The following describes what you need to do to get started using Snowflake as an Attunity Replicate target endpoint.

- Sign up for an Amazon Web Services account. Then use the AWS Management Console to purchase Snowflake On Demand - Standard or Snowflake On Demand - Premier and launch a Snowflake cluster.

  Make a note of the basic information about your AWS account and your Snowflake cluster, such as your password and user name. You will need this information to configure Attunity Replicate to work with the Snowflake data warehouse. For more information, see Setting General Connection Parameters.
- **Attunity Replicate for Windows:** Download and install the latest Windows 64-bit ODBC driver for Snowflake.
- **Attunity Replicate for Linux:** Download and install the latest Linux 64-bit ODBC driver for Snowflake.

For more information on setting up Snowflake as an AWS service, see https://docs.snowflake.net/manuals/index.html

Sign up for an Amazon S3 Bucket

You need to have an Amazon S3 bucket, preferably (for best performance) located in your Snowflake cluster region.

You must be able to access your Amazon S3 bucket directly from the Replicate machine.

For information on signing up for Amazon S3, see http://aws.amazon.com/s3/.

- **Bucket access credentials:** Make a note of the bucket name, region, access key and secret access key - you will need to provide them in the Attunity Replicate Snowflake target settings.
- **Bucket access permissions:** Attunity Replicate requires read/write/delete permissions to the Amazon S3 bucket.

Open the Required Firewall Port

Firewall port 443 needs to be opened for outbound communication.

Snowflake Data Types

The Snowflake database for Attunity Replicate supports most Snowflake data types. The following table shows the Snowflake target data types that are supported when using Attunity Replicate and the default mapping from Attunity Replicate data types.

Note Snowflake does not support applying changes to binary data types in **Batch optimized apply** mode. For more information on **Batch optimized apply** mode, see Changes Processing Tuning.

For information on how to view the data type that is mapped from the source, see the section for the source database you are using. For additional information about Attunity Replicate data types, see Replicate Data Types.

Table 9.17 | Supported Snowflake Data Types with Mapping from Attunity Replicate Data Types

Attunity Replicate Data Types	Snowflake Data Types
BOOL	BOOLEAN
BYTES	If length is => 1 and =< 32767, then: BINARY (Length in Bytes) If length is => 32768 and =< 2147483647, then: BINARY (65535)
DATE	DATE
TIME	VARCHAR (20)
TIMESTAMP	If scale is => 0 and =< 6, then: TIMESTAMP If scale is => 7 and =< 9, then: VARCHAR (37)
INT1	BYTEINT
INT2	SMALLINT
INT4	INTEGER
INT8	BIGINT
NUMERIC	If scale is => 0 and =< 37, then: NUMBER (p,s) If scale is => 38 and =< 127, then: NUMBER (Length)
REAL4	FLOAT4

Table 9.17 | Supported Snowflake Data Types with Mapping from Attunity Replicate Data Types (Cont.)

Attunity Replicate Data Types	Snowflake Data Types
REAL8	FLOAT8
STRING	If length is => 1 and =< 21845, then: VARCHAR (Length in Bytes) If length is => 21846 and =< 2147483647, then: VARCHAR (65535)
UINT1	BYTEINT
UINT2	INTEGER
UINT4	INTEGER
UINT8	BIGINT
WSTRING	If length is => 1 and =< 21845, then: VARCHAR (Length in Bytes) If length is => 21846 and =< 2147483647, then: VARCHAR (65535)

Note About Snowflake LOB support:

Full LOB data types are not supported. For information on including Limited-size LOB data types in the replication, see the Metadata tab description in Customizing Tasks .

BLOB	If length is => 1 and =< 32767, then: VARCHAR (Length in Bytes) If length is => 32768 and =< 2147483647, then: VARCHAR (65535)
NCLOB	NVARCHAR (Max LOB Size)
CLOB	VARCHAR (Max LOB Size)

Setting General Connection Parameters

This section describes how to configure general connection properties. For an explanation of how to configure advanced connection properties, see **Setting Advanced Connection Properties** below.

To add a Snowflake target endpoint to Attunity Replicate:

1. In the Attunity Replicate Console, click Manage Endpoint Connections to open the **Manage Endpoints Connections** dialog box.
2. In the **Manage Endpoint Connections** dialog box, click **New Endpoint Connection**.
3. In the **Name** field, type a name for your Snowflake data warehouse [service]. This can be any name that will help to identify your Snowflake database.
4. Optionally, in the **Description** field, type a description that helps to identify the Snowflake target database.
5. Select **TARGET** as the **role**.
6. Select **Snowflake** as the **Type**.
7. Enter the following **Snowflake target** information:
   - **Snowflake URL:** Your URL for accessing Snowflake on AWS.
   - **User name**: Your Snowflake user name.
   - **Password**: The password for the user entered in the **User name** field.
   - **Warehouse:** The name of your snowflake data warehouse.
   - **Database name**: Type the **database name** or click the browse button to select one from the available list.

   The information for these properties is available from the account page for Amazon Web Services (AWS) with the Snowflake cluster. If you do not have these values, refer to your AWS account or the Snowflake System Administrator for your enterprise.
8. Enter the following **Amazon S3 staging** information. You may need to click the **Amazon S3 staging** header to see the information.
   - **Bucket name**: Type the name of the Amazon S3 bucket where you are copying files to.
   - **Bucket region**: Select the Amazon S3 region where the S3 buckets and folders you are using are hosted. The default value is **US East (N. Virginia)**.

     Note: The bucket region specified must be the same region where your Snowflake database is located.
   - **Access key**: Type the access key information for Amazon S3.
   - **Secret key**: Type the secret key information for Amazon S3.
   - **Folder**: Type or browse to the S3 folder where you are copying files to.

   The information for these properties is available from your Amazon Web Services (AWS) account. If you do not have these values, refer to your AWS account or the Snowflake System Administrator for your enterprise

Note

- This information is case sensitive.
- To determine if you are connected to the database you want to use or if the connection information you entered is correct, click **Test Connection**.

If the connection is successful a message in green is displayed. If the connection fails, an error message is displayed at the bottom of the dialog box.

To view the log entry if the connection fails, click **View Log**. The server log is displayed with the information for the connection failure. Note that this button is not available unless the test connection fails.

Using Advanced Properties for an Snowflake Target

In the **Advanced** tab, you can set the following parameters:

- **Max file size (MB)**: Select or type the maximum size of the CSV file used to transfer data to Snowflake. The default value is 100 MB.
- **ODBC driver**: The name of the default ODBC driver you are using to connect to Snowflake. The default value is **SnowflakeDSIIDriver**.
- **Additional ODBC connection properties**: Type any additional ODBC connection properties if required.

Internal Parameters

Internal parameters are parameters that are not exposed in the UI. You should only use them if instructed by Attunity Support.

To add internal Attunity Replicate parameters:

1. Click the **Internal Parameters** link.

   The **Internal Parameters** dialog box opens.
2. In the edit box, type the name of the parameter you need to add and then click it.
3. The parameter is added to the table below the search box with its default value.
4. Change the default value as required.
5. To reset the parameter value to its default, click the "Restore default value" icon at the end of the row.

Settings Summary

You can view a summary of your settings by clicking the **Setting Summary** link. This is useful if you need to send a summary of your settings to Attunity Support.

Using Amazon S3 as a Target

This chapter describes how to set up and use Amazon S3 as a target in a replication task.

In this section:

Prerequisites

Before you can use Amazon S3 as a target endpoint in a Replicate task, the following prerequisites must be met:

- You must have an Amazon S3 bucket that is accessible from the Replicate Server machine.

  For information on signing up for Amazon S3, see http://aws.amazon.com/s3/.
- **Bucket access credentials:** Make a note of the bucket name, region, access key and secret access key - you will need to provide them in the Attunity Replicate Amazon S3 target settings.
- **Bucket access permissions:** Attunity Replicate requires the following bucket access permissions:

```
{
        "Version": "2012-10-17",
        "Statement": [
            {
             "Sid": "Stmt1497347821000",
             "Effect": "Allow",
             "Action": [
                 "s3:GetBucketLocation",
                 "s3:ListBucket"
             ],
             "Resource": [
                 "arn:aws:s3:::YOUR_BUCKET_NAME"
             ]
          },
          {
             "Sid": "Stmt1497344984000",
```

```
                "Effect": "Allow",
                "Action": [
                    "s3:PutObject",
                    "s3:GetObject",
                    "s3:DeleteObject"
                ],
                "Resource": [
                    "arn:aws:s3:::YOUR_BUCKET_NAME/target_path",
                    "arn:aws:s3:::YOUR_BUCKET_NAME/target_path/*"
                ]
            }
        ]
    }
```

Where `YOUR_BUCKET_NAME` is the name of your bucket and `target_path` is the intended location of the target files in your bucket.

Note If the target path is the bucket root, just specify `"/target_path"` with an empty string.

Amazon S3 Target Overview

When using Amazon S3 as a target in a Replicate task, both the Full Load and CDC data are written to data files. Depending on the endpoint settings, data files can be either CSV or JSON files. While the explanations in this section relate to CSV files, the same is true for JSON files

Full Load files are named using incremental counters e.g. LOAD00001.csv, LOAD 00002.csv, etc. whereas Apply Changes files are named using timestamps e.g. 20141029-1134010000.csv.

Note When the Create metadata files in the target folder option is enabled, a corresponding metadata file is created using the same naming format, but with a **.dfm** extension.

For each source table, a folder is created in the specified Amazon S3 bucket. The data files are created on the Replicate Server machine and are then uploaded to the specified Amazon S3 bucket once the specified maximum file size (Full Load) has been reached and/or the Change Processing (CDC) upload conditions have been met.

DDL Handling

When a DDL change is captured, Replicate will close the data file and also create a DFM file if the **Create metadata files in the target folder** option is enabled. When the next batch of changes arrives, Replicate will create a new data file containing the changes. Note that the DFM file created for the new data file will match the new table structure.

Limitations

The following limitations apply to the Amazon S3 target endpoint:

- Only the following DDLs are supported: Truncate table, Drop table, Create table, Add Column, Rename Column, Drop Column, and Convert Data Type.
- Full LOB Mode is not supported
- UPDATE and DELETE statements are not supported in Apply Changes replication mode
- Batch Optimized Apply mode is not supported
- Target lookup is not supported
- The <target folder> parameter cannot include special characters

Amazon S3 Target Data Types

The following table shows the default mapping from Attunity Replicate data types to Amazon S3 target data types. Note that the data type mapping is only relevant if the Create metadata files in the target folder option is enabled.

For information on source data type mappings, see the section for the source endpoint you are using.

For additional information about Attunity Replicate data types, see Replicate Data Types.

Table 9.18 | Supported Amazon S3 Target Data Types with Mapping from Attunity Replicate Data Types

Attunity Replicate Data Types	Amazon S3 Target Data Types
DATE	DATE
TIME	TIME
DATETIME	DATETIME
BYTES	BYTES (length)
BLOB	BLOB
REAL4	REAL4 (7)
REAL8	REAL8 (14)
INT1	INT1 (3)
INT2	INT2 (5)
INT4	INT4 (10)
INT8	INT8 (19)
UINT1	UINT1 (3)
UINT2	UINT2 (5)

Table 9.18 | Supported Amazon S3 Target Data Types with Mapping from Attunity Replicate Data Types (Cont.)

Attunity Replicate Data Types	Amazon S3 Target Data Types
UINT4	UINT4 (10)
UINT8	UINT8 (20)
NUMERIC	NUMERIC (p,s)
STRING	STRING (Length)
WSTRING	STRING (Length)
CLOB	CLOB
NCLOB	NCLOB
BOOLEAN	BOOLEAN (1)

Setting General Connection Properties

This section describes how to configure general connection properties. For an explanation of how to configure advanced connection properties, see **Setting Advanced Connection Properties** below.

To add an Amazon S3 target endpoint to Attunity Replicate:

1. In **Tasks** view, click **Manage Endpoint Connections** to open the **Manage Endpoints Connections** dialog box. Then click the **New Endpoint Connection** button. For more information on adding an endpoint to Attunity Replicate, see Working with Endpoints.
2. In the **Name** field, type a name for your endpoint. This can be any name that will help to identify the endpoint being used.
3. Optionally, in the **Description** field, type a description that helps to identify the endpoint.
4. Select **TARGET** as the endpoint **role**.
5. Select **Amazon S3** as the endpoint **Type**.
6. Configure the remaining settings in the **General** tab as described in the table below.

Table 9.19 | Amazon S3 Target Endpoint - General Tab Options

Option	Description
Amazon S3 Storage	
Bucket name	Enter the name of your Amazon S3 bucket.
Bucket region	Select the Amazon S3 region where your bucket is located.

Table 9.19 | Amazon S3 Target Endpoint - General Tab Options (Cont.)

Option	Description
Access options	Choose one of the following: » Key pair Choose this method to authenticate with your Access Key and Secret Key. » IAM Roles for EC2. Choose this method if the machine on which Attunity Replicate is installed is configured to authenticate itself using an IAM role. For information on IAM roles, see: http://docs.aws.amazon.com/IAM/latest/UserGuide/id_roles.html
Access key	If you selected **Key pair** as your access method, enter the access key information for Amazon S3.
Secret key	If you selected **Key pair** as your access method, enter the secret key information for Amazon S3.
Target folder	Enter the target folder in your Amazon S3 bucket.
File Attributes Delimiters can be standard characters or a hexadecimal (hex) value. Note that the "`0x`" prefix must be used to denote a hexadecimal delimiter (e.g. `0x01 = SOH`). In the **Field delimiter**, **Record delimiter** and **Null value** fields, the delimiter can consist of concatenated hex values (e.g. `0x0102 = SOHSTX`), whereas in the **Quote character** and **Escape character** fields, it can only be a single hex value. **Note** The hexadecimal number `0x00` is not supported (i.e. only `0x01-0xFF` are supported).	
Format	You can choose to create the target files in **CSV** or **JSON** format. In a JSON file, each record is represented by a single line, as in the following example: { "book_id": 123, "title": "Alice in Wonderland", "price": 6.99, "is_hardcover": false } { "book_id": 456, "title": "Winnie the Pooh", "price": 6.49, "is_hardcover": true } { "book_id": 789, "title": "The Cat in the Hat", "price": 7.23, "is_hardcover": true } **Note** If you choose **JSON** format , the following fields will be hidden as they are only relevant to CSV format: **Field delimiter**, **Record delimiter**, **Null value**, **Quote character**, **Escape character**, and **Add metadata header**.
Field delimiter	The delimiter that will be used to separate columns in the target files.

Table 9.19 | Amazon S3 Target Endpoint - General Tab Options (Cont.)

Option	Description
	The default is a comma. **Example using a comma as a delimiter:** mike,male
Record delimiter	The delimiter that will be used to separate records (rows) in the target files. The default is a newline (`\n`). **Example using an asterisk as a delimiter:** mike,male*sara,female
Null value	The string that will be used to indicate a null value in the target files. **Example (where * is the row delimiter and @ is the null value):** mike,male,295678*sara,female,@
Quote character	The character that will be used at the beginning and end of a column. The default is the double-quote character ("). When a column that contains column delimiters is enclosed in double-quotes, the column delimiter characters are interpreted as actual data, and not as column delimiters. **Example (where a comma is the column delimiter):** "sunroof, power-steering"
Escape character	The character used to escape a quote character in the actual data. **Example (where" is the quote character and \ is the escape character):** 1955,"old, \"rare\", Chevrolet",$1000
Add metadata header	You can optionally add a header row to the data files. The header row can contain the source column names and/or the intermediate (i.e. Replicate) data types. Example of a target file with a header row when both **With column names** and **With data types** are selected: Position:DECIMAL(38,0),Color:VARCHAR(10) 1,"BLUE" 2,"BROWN" 3,"RED" ...
Maximum file size	The maximum size a file can reach before it is closed (and optionally compressed). This value applies both to data files and to Reference Files. For information on generating reference files, see Generate reference files (locally).

Table 9.19 | Amazon S3 Target Endpoint - General Tab Options (Cont.)

Option	Description
Compress files using	Choose **GZIP** to compress the target files or **NONE** (the default) to leave them uncompressed.
Change Processing	
Apply/Store changes when:	
File size reaches	Specify the maximum size of Change Data to accumulate before uploading the file to Amazon S3.
Elapsed time reaches	Specify the maximum time to wait before applying the changes.
Metadata Files	
Create metadata files in the target folder	When this option is selected, for each data file, a matching metadata file with a **.dfm** extension will be created under the specified target folder. The metadata files (which are in standard JSON format) provide additional information about the task/data such as the source endpoint type, the source table name, the number of records in the data file, and so on. For a full description of the metadata file as well as possible uses, see Metadata File Description .
Data Encryption	
Encryption options	Choose one of the following: » Server-Side Encryption with Amazon S3-Managed Keys (SSE-S3). This is the default. » Server-Side Encryption with AWS KMS-Managed Keys (SSE-KMS) This option also requires you to specify your **KMS Key ID**. For more information on the available sever-side encryption methods, see: http://docs.aws.amazon.com/AmazonS3/latest/dev/serv-side-encryption.html » None

7. To determine if the connection information you entered is correct, click **Test Connection**. If the connection test is successful, click **Save**.

Note As part of connection testing process, Replicate uploads a test file to the specified Amazon S3 Target folder and then deletes it once a connection has been established.

If the connection is successful a message in green is displayed. If the connection fails, an error message is displayed at the bottom of the dialog box.

To view the log entry if the connection fails, click **View Log**. The server log is displayed with the information for the connection failure. Note that this button is not available unless the test connection fails.

Setting Advanced Connection Properties

In the **Advanced** tab, you can enable the creation of reference files and set post-processing actions. These options are described in detail below.

Table 9.20 | Amazon S3 Target Endpoint - Advanced Tab Options

Option	Description
Generate reference files (locally)	Select this option to generate a Reference File (on Replicate Server) containing the full path to the Apply Changes data files. **Note** The reference file only points to the location of the Apply Changes files, and not the Full Load files.
Reference file folder	The folder on the Replicate machine in which the Reference File will be created. **Example:** c:\temp\
Post-process files	You can process the final target files using a custom command. The command will be run whenever a data file is created. **Note** If the **Generate a reference file** option is selected, a row (specifying the file's location) will be added to the Reference File only *after* the command completes successfully. » **Command name** - The location of the command e.g. `C:\utils\move.exe`. » **Working directory** - The directory where you want the command to run. » **Parameters** - Specify any parameters that need to be passed to the command during runtime. You can use the following built-in parameters: `${FILENAME}` - The full path to the CSV file containing the full load or CDC data. `${METADATA_FILENAME}` - The full path to the DFM file containing the metadata. For information on creating metadata files, see Setting General Connection Properties.

Table 9.20 | Amazon S3 Target Endpoint - Advanced Tab Options (Cont.)

Option	Description
	Note If the CSV/DFM file paths contain spaces, you must enclose these parameters with quotation marks (e.g `"${FILENAME}"`).

Standard Post Command Exit Codes

The post-processing command must return a proper exit code. You can either use the standard exit code values described below or set a custom exit code value as described in Setting Post Command Exit Codes with an Internal Parameter.

- **0** - Success
- **1** - Recoverable error. The task will recover from the point of failure according to the settings in the Environmental Errors tab.
- **2** - Table error. If a table error occurs, Replicate will handle the error according to the settings in the Table Errors tab.
- **3** (or any other value e.g. -100) - Fatal error. The task will fail and not attempt recovery.

Setting Post Command Exit Codes with an Internal Parameter

You can use internal parameters to set exit codes with custom values. This is especially useful if your application already uses the standard exit code values.

See Standard Post Command Exit Codes above for a description of the exit codes.

- `successExitCode`
- `recoverableErrorExitCode`
- `tableErrorExitCode`
- `fatalErrorExitCode`

For instructions on setting internal parameters, see Internal Parameters.

Internal Parameters

Internal parameters are parameters that are not exposed in the UI and should only be used if instructed by Attunity Support.

To add internal Attunity Replicate parameters:

1. Click the **Internal Parameters** link.
   The **Internal Parameters** dialog box opens.
2. In the edit box, type the name of the parameter you need to add and then click it.
3. The parameter is added to the table below the search box with its default value.
4. Change the default value as required.
5. To reset the parameter value to its default, click the "Restore default value" icon at the end of the row.

Settings Summary

You can view a summary of your setting by clicking the View Setting Summary link. This is useful if you need to send a summary of your setting to Attunity Support.

Generating Reference Files

In the **Advanced** tab of the Amazon S3 target endpoint, you can enable the **Generate a reference file** option. The Reference File contains a list of the Change File locations and is therefore only relevant if the task's Apply Changes or Store Changes options are enabled.

The format of the reference file name is as follows:

```
<amazon_s3_target_endpoint_display_name><counter>.csv|json
```

Example:

```
AmazonS300000001.csv
```

Note The counter suffix increases incrementally each time a new Reference File is generated (i.e. when the file reaches the maximum size defined in the **General** tab). Once a new Reference File has been generated, you can delete the old reference file(s) if required.

Whenever an Apply Changes data file is created, a new row is added to the Reference File in the following format:

```
<Source_Table_Name>,<bucket_name>/<path>/<file_name>
```

Example:

```
employees,bigdata/new/files/my.company/20170611-120144192.csv
```

Note that if the **Post-process files** option in the **Advanced** tab is also enabled, the Reference File will be generated *after* the post-processing completes.

Using HP Vertica as a Target

This section describes how to set up and use an HP Vertica database as a target database in a replication task.

In this section:

Prerequisites

The following section describes the prerequisites for working with Attunity Replicate on Windows or Linux and the HP Vertica target database.

Replicate Server for Windows

The following section describes the steps you need to perform to work with Attunity Replicate for Windows and HP Vertica as a target database in a Replicate task:

- HP Vertica ODBC 64-bit client installed on the computer where Attunity Replicate is located.
- VSQL CLI Client installed on the computer where Attunity Replicate is located.

Replicate Server for Linux

The following section describes the steps you need to perform to work with Attunity Replicate for Linux and HP Vertica as a target database in a Replicate task:

1. On the Attunity Replicate machine, install the HP Vertica client for Linux:

   ```
   vertica-client-<version>.x86_64
   ```

   Example:

   ```
   vertica-client-7.0.0-0.x86_64
   ```

   Note HP Vertica 7.1 client is not compatible with database versions earlier than HP Vertica 7.1.

2. Makes sure that the `/etc/odbcinst.ini` file contains the following entry for HP Vertica, as in the following example:

   ```
   [Vertica]
   Driver = /opt/vertica/lib64/libverticaodbc.so
   ```

```
DriverODBCVer = 3.0
UsageCount = 1
```

Security Requirements

You must provide HP Vertica account access to the Attunity Replicate user. The Replicate user must also have the following privileges in the HP Vertica database:

- CREATE TABLE Privileges:
  - CREATE privilege on schema
- DROP TABLE Privileges:
  - USAGE privilege on the schema that contains the table or schema owner
- TRUNCATE Privileges (If the task is configured to truncate existing tables):
  - USAGE privilege on the schema that contains the table or schema owner
- ALTER TABLE (ADD/DROP/ RENAME/ALTER-TYPE COLUMN) Privileges:
  - USAGE privilege on the schema that contains the table or schema owner
- INSERT Privileges:
  - INSERT privilege on table
  - USAGE privilege on the schema that contains the table
- UPDATE Privileges:
  - UPDATE privilege on table
  - USAGE privilege on the schema that contains the table
  - SELECT privilege on the table when executing an UPDATE statement that references table column values in a WHERE or SET clause
- DELETE Privileges:
  - DELETE privilege on table
  - USAGE privilege on schema that contains the table
  - SELECT privilege on the table when executing a DELETE statement that references table column values in a WHERE or SET clause

HP Vertica Target Data Types

The HP Vertica database for Attunity Replicate supports most HP Vertica data types. The following table shows the HP Vertica target data types that are supported when using Attunity Replicate and the default mapping from Attunity Replicate data types.

Note HP Vertica does not support applying changes to binary data types in **Batch optimized apply** mode. For more information on **Batch optimized apply** mode, see Changes Processing Tuning.

For information on how to view the data type that is mapped from the source, see the section for the source database you are using.

For additional information about Attunity Replicate data types, see Replicate Data Types.

Table 9.21 | Supported HP Vertica Data Types with Mapping from Attunity Replicate Data Types

Attunity Replicate Data Types	HP Vertica Data Types
BOOLEAN	BOOLEAN
BYTES	VARBINARY (Length)
DATE	DATE
TIME	TIME (p)
DATETIME	TIMESTAMP
INT1	INTEGER
INT2	INTEGER
INT4	INTEGER
INT8	INTEGER
NUMERIC	NUMERIC (p,s)
REAL4	FLOAT
REAL8	FLOAT
STRING	VARCHAR (Length)
UINT1	INTEGER
UINT2	INTEGER
UINT4	INTEGER
UINT8	INTEGER
WSTRING	VARCHAR (Length)
BLOB	VARBINARY (65,000)
CLOB	VARCHAR (65,000)
NCLOB	VARCHAR (65,000)

Setting General Connection Properties

This section describes how to configure general connection properties. For an explanation of how to configure advanced connection properties, see **Setting Advanced Connection Properties** below.

To add an HP Vertica target endpoint to Attunity Replicate:

1. In **Tasks** view, click **Manage Endpoint Connections** to open the **Manage Endpoints Connections** dialog box. Then click the **New Endpoint Connection** button.

2. In the **Name** field, type a name for your database. This can be any name that will help to identify the database being used.
3. In the **Description** field, type a description that helps to identify the HP Vertica database. This is optional.
4. Select **TARGET** as the database **role**.
5. Select **HP Vertica** as the database **Type**.
6. In the **Server** field, enter the name of the HP Vertica server.
7. Optionally, change the default **Port** (5433).
8. Type the HP Vertica authentication information (**User Name**, **Password**) for the authorized user for this HP Vertica database. If you do not know this information, see your HP Vertica database Administrator (DBA).

   Note
   - This information is required. If you are using the **Advanced** tab to create a custom string, make sure to include the **User Name** and **Password** properties. See Setting Advanced Connection Properties for more information.
   - This information is case sensitive.
   - If you want to set custom properties for this database, see Setting Advanced Connection Properties.

   Important: Make sure that the HP Vertica user entered in the HP Vertica Authentication section has the correct access privileges. For information on how to provide the required privileges, see Security Requirements.]
9. In the **Database name** field, enter the name of the HP Vertica database.

Setting Advanced Connection Properties

In the **Advanced** tab, you can set the following parameters:

- **Max file size**: Select or type the maximum size (in KB) of a CSV file before the file is loaded into the HP Vertica database. The default value is 32000 KB.

Internal Parameters

Internal parameters are parameters that are not exposed in the UI. You should only use them if instructed by Attunity Support.

To add internal Attunity Replicate parameters:

1. Click the **Internal Parameters** link.
   The **Internal Parameters** dialog box opens.
2. In the edit box, type the name of the parameter you need to add and then click it.
3. The parameter is added to the table below the search box with its default value.

4. Change the default value as required.
5. To reset the parameter value to its default, click the "Restore default value" icon at the end of the row.

Settings Summary

You can view a summary of your settings by clicking the **Setting Summary** link. This is useful if you need to send a summary of your settings to Attunity Support.

Using Microsoft APS PDW as a Target

This section describes how to set up and use Microsoft APS PDW as a target database in a replication task.

In this section:

Prerequisites

Note

- Attunity Replicate must be installed on any Windows computer in your network.
- A Microsoft APS PDW account with the required access privileges is required.

The following client components must be installed on the Attunity Replicate machine:

- SQL Server Native Client 11.0
- Microsoft SQL Server 2012 Parallel Data Warehouse Tools x64

Limitations

The following section describes the limitations of using Microsoft APS PDW as a Replicate target.

- Source columns with CHAR/VARCHAR data types and a non-Latin collation (e.g. "Chinese_PRC_CI_AS") need to be mapped to NVARCHAR. This can be done by defining a global transformation for all tables in the replication task or by defining a single transformation for a specific table.

  For more information on defining transformations, see Defining Global Transformations and Defining Transformations for a Single Table/View.
- Microsoft APS PDW does not support empty (NULL) columns. Consequently, when replicating a source column with an empty value, Replicate inserts a space into the corresponding target column.

Security Requirements

You must provide Microsoft APS PDW account access to the Attunity Replicate user. This user must have LOAD permission and applicable permissions (INSERT, UPDATE, DELETE)

on the destination table.

Microsoft APS PDW Target Data Types

The Microsoft APS PDW database for Attunity Replicate supports most Microsoft APS PDW data types. The following table shows the Microsoft APS PDW target data types that are supported when using Attunity Replicate and the default mapping from Attunity Replicate data types.

For information on how to view the data type that is mapped from the source, see the section for the source database you are using.

For additional information about Attunity Replicate data types, see Replicate Data Types.

Table 9.22 | Supported Microsoft APS PDW Data Types with Mapping from Attunity Replicate Data Types

Attunity Replicate Data Types	Microsoft APS PDW Data Types
BOOLEAN	BIT
BYTES	VARBINARY (Length)
DATE	DATE
TIME	TIME
DATETIME	DATETIME2 (scale)
INT1	TINYINT
INT2	SMALLINT
INT4	INTEGER
INT8	BIGINT
NUMERIC	DECIMAL (p,s)
REAL4	FLOAT (24)
REAL8	FLOAT (53)
STRING	VARCHAR (Length)
UINT1	TINYINT
UINT2	SMALLINT
UINT4	INTEGER
UINT8	BIGINT
WSTRING	NVARCHAR (Length)
BLOB	VARBINARY (8000)
NCLOB	NVARCHAR (4000)
CLOB	VARCHAR (8000)

Setting General Connection Properties

This section describes how to configure general connection properties. For an explanation of how to configure advanced connection properties, see **Setting Advanced Connection Properties** below.

To add a Microsoft APS PDW target endpoint to Attunity Replicate:

1. In **Tasks** view, click **Manage Endpoint Connections** to open the **Manage Endpoints Connections** dialog box. Then click the **New Endpoint Connection** button.
2. In the **Name** field, type a name for your database. This can be any name that will help to identify the database being used.
3. In the **Description** field, type a description that helps to identify the Microsoft APS PDW database. This is optional.
4. Select **TARGET** as the database **role**.
5. Select **Microsoft APS PDW** as the database **Type**.
6. In the **Server name** field, enter the hostname or IP address of the Microsoft APS PDW machine.
7. Optionally, change the default **Port** (5000).
8. Type the Microsoft APS PDW authentication information (**User Name**, **Password**) for the authorized user for this Microsoft APS PDW database. If you do not know this information, see your Microsoft APS PDW database Administrator (DBA).

   Note

   - This information is required. If you are using the **Advanced** tab to create a custom string, make sure to include the **User Name** and **Password** properties. See Setting Advanced Connection Properties for more information.
   - This information is case sensitive.
   - If you want to set custom properties for this database, see Setting Advanced Connection Properties.

   Important: Make sure that the Microsoft APS PDW user entered in the Microsoft APS PDW Authentication section has the correct access privileges. For information on how to provide the required privileges, see Security Requirements.

9. In the **Database name** field, enter the name of the Microsoft APS PDW database.

Setting Advanced Connection Properties

In the **Advanced** tab, you can set the following parameters:

- **Maximum file size**: Select or type the maximum size (in KB) of a CSV file before the file is loaded into the Microsoft APS PDW database. The default value is 32000 KB.

- **Create hash distribution**: Enabling this option will turn on the Microsoft APS PDW hash distribution function.
- **Additional ODBC connection properties**: Specify any additional ODBC connection parameters that you want to use.

Internal Parameters

Internal parameters are parameters that are not exposed in the UI. You should only use them if instructed by Attunity Support.

To add internal Attunity Replicate parameters:

1. Click the **Internal Parameters** link.

   The **Internal Parameters** dialog box opens.
2. In the edit box, type the name of the parameter you need to add and then click it.
3. The parameter is added to the table below the search box with its default value.
4. Change the default value as required.
5. To reset the parameter value to its default, click the "Restore default value" icon at the end of the row.

Settings Summary

You can view a summary of your settings by clicking the **Setting Summary** link. This is useful if you need to send a summary of your settings to Attunity Support.

Using ODBC to Connect to a Target

This section describes how to use ODBC connectivity to connect to a target endpoint.

Note When using HP NonStop SQL/MP (ARC) as an ODBC target, several additional procedures must be performed. For a detailed explanation, see Using HP NonStop SQL/MP as an ODBC Target.

In this section:

Using ODBC to Connect to a Target

The following topics describe what you need to use an ODBC endpoint as a target endpoint in an Attunity Replicate task.

Note When using HP NonStop SQL/MP (ARC) as an ODBC target, several additional procedures must be performed. For a detailed explanation, see Using HP NonStop SQL/MP as an ODBC Target.

ODBC Target Data Types

The following table shows the ODBC target data types that are supported when using Attunity Replicate and the default mapping from Attunity Replicate data types.

Note ODBC does not support applying changes to binary data types in **Batch optimized apply** mode. For more information on **Batch optimized apply** mode, see Changes Processing Tuning.

For information on how to view the data type that is mapped from the source, see the section for the source endpoint you are using.

For additional information about Attunity Replicate data types, see Replicate Data Types.

Table 9.23 | Supported ODBC Target Data Types with Mapping from Attunity Replicate Data Types

Attunity Replicate Data Types	ODBC Data Types
BOOLEAN	**SQL_BIT**
BYTES	**SQL_VARBINARY**
DATE	**SQL_TYPE_DATE**
TIME	**SQL_TYPE_TIME**
DATETIME	**SQL_TYPE_TIMESTAMP**
INT1	**SQL_SMALLINT**
INT2	**SQL_SMALLINT**
INT4	If the target endpoint supports precision and scale then: **SQL_INTEGER** Otherwise: **SQL_VARCHAR**
INT8	**SQL_BIGINT**
NUMERIC	**SQL_NUMBER**
REAL4	**SQL_REAL**
REAL8	**SQL_DOUBLE**
STRING	**SQL_VARCHAR**
UINT1	**SQL_TINYINT**
UINT2	**SQL_SMALLINT**
UINT4	**SQL_INTEGER**
UINT8	**SQL_BIGINT**
WSTRING	**SQL_WVARCHAR**

Note If the target endpoint does not support ODBC data types, data types are mapped to SQL_VARCHAR.

Setting General Connection Properties

This section describes how to configure general connection properties. For an explanation of how to configure advanced connection properties, see **Setting Advanced Connection Properties** below.

To add an ODBC endpoint to Attunity Replicate:

1. In **Tasks** view, click **Manage Endpoint Connections** to open the **Manage Endpoints Connections** dialog box. Then click the **New Endpoint Connection** button.
2. In the **Name** field, type a name for your ODBC endpoint. This can be any name that will help to identify the endpoint being used.
3. In the **Description** field, type a description that helps to identify the ODBC endpoint. This is optional.
4. Select **TARGET** as the endpoint **role**.
5. Select **ODBC** as the endpoint **Type**.
6. Select *one* of the following:
   - **DSN**: Select this to connect to an ODBC-supported endpoint using a DSN. When you select DSN you must select the DSN you are using from the list.

     Note When connecting to SQL/MP, you must use a connection string, which should include the name of the Replicate ARC Unicode ODBC driver. See Connection String for an example.

     If the DSN you want to use is not included in the list, make sure that the endpoint client is installed on the computer with Attunity Replicate and that the DSN is defined. Note that the ODBC provider client must be 64-bit. For more information, see Prerequisites .

     Note If you are using an ARC CDC Agent as the source in a Replicate task, you cannot select the DSN for the Attunity ODBC driver as the target. In this case, to use Attunity ODBC as a target, you must enter the connection string manually by selecting **Connection String** and following the directions for that option in this procedure.

   - **Connection String**: Select this to connect to an ODBC-supported endpoint using a connection string then type a valid connection string in the field below. For information on how to create a connection string, see the documentation for the ODBC endpoint provider you are using.

     Example of an SQL/MP Connection String:

     ```
     Driver={Attunity Replicate ARC ODBC Driver 3.5 (Unicode)};BindUrl=attconnect://ais_server_ip:ais_server_port/ais_workspace;DefTdpName=ais_target_datasource_name;OneTdpMode=1;qptdpname=BINDURL1;queryProcessor/noThreads=true;}}
     ```

     Note that if you specify a password in your connection string, it will be revealed as plain text in the task log files. It is therefore recommended to specify the password in the GUI **Password** field.

Note

- You can use the **Advanced** tab to add specific properties and create a custom connect string. In this case, you do not need to enter information in this tab. For more information on using the **Advanced** tab, see Setting Advanced Connection Properties.
- To determine if you are connected to the endpoint you want to use or if the connection information you entered is correct, click **Test Connection**.

  If the connection is successful a message in green is displayed. If the connection fails, an error message is displayed at the bottom of the dialog box.

  To view the log entry if the connection fails, click **View Log**. The server log is displayed with the information for the connection failure. Note that this button is not available unless the test connection fails.

7. Type the authentication information (**User Name**, **Password**) for the authorized user for the ODBC endpoint being used. For example, the IBM DB2 system administrator if you are using a IBM DB2 provider. If you do not know this information, see your ODBC Endpoint System Administrator.

Note

- When you select **Connection String** be sure to include **User name/password** information in the connection string that you type in the box.

  If you are using the **Advanced** tab to create a custom string, make sure to include the **User Name** and **Password** properties. For more information, see Setting Advanced Connection Properties.
- This information is case sensitive.
- You can set custom properties in the **Advanced** tab. For more information, see Setting Advanced Connection Properties.

Important: Make sure that the ODBC endpoint user has the correct access privileges for the ODBC provider being used.

Setting Advanced Connection Properties

In the **Advanced** tab, you can set the following properties:

- **Provider syntax**: Select the name of the provider syntax. Note that when replicating to an HP NonStop SQL/MP target, you must select **SQLMP (ARC)** as the provider type.
- **Load using CSV**: Select to load the data using a CSV file.
- **Max file size** (KB): Select or type the maximum size (in KB) of a CSV file before the file is moved into the load folder. The default value is 32000 KB.

Internal Parameters

Internal parameters are parameters that are not exposed in the UI. You should only use them if instructed by Attunity Support.

To add internal Attunity Replicate parameters:

1. Click the **Internal Parameters** link.

   The **Internal Parameters** dialog box opens.
2. In the edit box, type the name of the parameter you need to add and then click it.
3. The parameter is added to the table below the search box with its default value.
4. Change the default value as required.
5. To reset the parameter value to its default, click the "Restore default value" icon at the end of the row.

Settings Summary

You can view a summary of your settings by clicking the **Setting Summary** link. This is useful if you need to send a summary of your settings to Attunity Support.

Using Microsoft Azure SQL Data Warehouse as a Target

This section describes how to set up and use Microsoft Azure SQL Data Warehouse as a target in a replication task. Microsoft Azure SQL Data Warehouse is located in the cloud and is accessed through your Microsoft Azure account.

In this section:

Overview

In the first stage of the replication process, Attunity Replicate moves the data files created by the source database into Microsoft Azure Blob Storage where thy are stored as CSV files. The files are then loaded into the proper tables in the Microsoft Azure SQL Data Warehouse data warehouse (using PolyBase).

The Replicate Microsoft Azure SQL Data Warehouse database provides full automation for:

- Schema generation and data type mapping
- Full load of source database tables
- Incremental load of changes made to source tables
- Application of schema changes (DDL) made to the source tables.
- Synchronization between full load and CDC processes.

Manual control is also available if needed.

Limitations

The following section describes the limitations of using Microsoft Azure SQL Data Warehouse as a Replicate target.

- Source columns with CHAR/VARCHAR data types and a non-Latin collation (e.g. "Chinese_PRC_CI_AS") need to be mapped to NVARCHAR. This can be done by defining a global transformation for all tables in the replication task or by defining a single transformation for a specific table.

  For more information on defining transformations, see Defining Global Transformations and Defining Transformations for a Single Table/View.
- Microsoft Azure SQL Data Warehouse does not support empty (NULL) columns. Consequently, when replicating a source column with an empty value, Replicate inserts a space into the corresponding target column.
- The rename column DDL is not supported.

Microsoft Azure SQL Data Warehouse Endpoint Prerequisites

The following sections describe the prerequisites necessary for using Microsoft Azure SQL Data Warehouse as a target endpoint in a Replicate task.

- Sign up for Microsoft Azure Blob Storage
- Sign up for Microsoft Azure SQL Data Warehouse
- Open the Required Firewall Ports
- Install the Required Client

Sign up for Microsoft Azure Blob Storage

Sign up for an Azure Blob Storage account and make a note of the account name, account key, container name and target folder - you will need to provide them later.

Note For best performance, the Azure Blob Storage container should be in the same region as your Microsoft Azure SQL Data Warehouse.

Required Permissions

Attunity Replicate performs the following operations on the Azure Blob Storage container/folder:

- On the Azure Blob Storage container: LIST
- On the Azure Blob Storage folder: READ, WRITE and DELETE

The user specified in the Microsoft Azure SQL Data Warehouse endpoint settings must be granted the above permissions.

Sign up for Microsoft Azure SQL Data Warehouse

Sign up for Microsoft Azure SQL Data Warehouse and make a note of the server name, port, user name, password, database name and Azure Blob Storage access Credential - you will need to provide them later. Note that if you have not already created an Azure Blob Storage access Credential, you can configure Replicate to create one automatically as described in Setting General Connection Properties.

Required Permissions

Attunity Replicate performs the following operations on the replicated tables within Microsoft Azure SQL Data Warehouse:

- SELECT, INSERT, UPDATE and DELETE
- Bulk Load
- CREATE, ALTER, DROP (if required by the task's definition)

Unless the user is the DB Owner, the user specified in the Microsoft Azure SQL Data Warehouse endpoint settings must be granted the above permissions.

Open the Required Firewall Ports

Open firewall port 1433 (Microsoft Azure SQL Data Warehouse) for outbound communication.

Install the Required Client

Install SQL Server Native Client 11 (for connecting to Microsoft Azure SQL Data Warehouse) on the Attunity Replicate machine.

Microsoft Azure SQL Data Warehouse Data Types

The Microsoft Azure SQL Data Warehouse database for Attunity Replicate supports most Microsoft Azure SQL Data Warehouse data types. The following table shows the Microsoft Azure SQL Data Warehouse target data types that are supported when using Attunity Replicate and the default mapping from Attunity Replicate data types.

For information on how to view the data type that is mapped from the source, see the section for the source database you are using. For additional information about Attunity Replicate data types, see Replicate Data Types.

Table 9.24 | Supported Microsoft Azure SQL Data Warehouse Data Types with Mapping from Attunity Replicate Data Types

Attunity Replicate Data Types	Microsoft Azure SQL Data Warehouse Data Types
BOOL	BIT
BYTES	If length is => 1 and =< 8000, then: VARBINARY (Length in Bytes)

Table 9.24 | Supported Microsoft Azure SQL Data Warehouse Data Types with Mapping from Attunity Replicate Data Types (Cont.)

Attunity Replicate Data Types	Microsoft Azure SQL Data Warehouse Data Types
	If length is => 8001 and =< 2147483647, then: VARBINARY (8000)
DATE	DATE
TIME	TIME
DATETIME	DATETIME2 (s)
INT1	TINYINT
INT2	SMALLINT
INT4	INTEGER
INT8	BIGINT
NUMERIC	DECIMAL (p,s)
REAL4	FLOAT(24)
REAL8	FLOAT(53)
STRING	If length is => 1 and =< 8000, then: VARCHAR (Length in Bytes) If length is => 8001 and =< 2147483647, then: VARCHAR (8000)
UINT1	TINYINT
UINT2	SMALLINT
UINT4	INTEGER
UINT8	BIGINT
WSTRING	If length is => 1 and =< 4000, then: NVARCHAR (Length in Bytes) If length is => 4001 and =< 2147483647, then: NVARCHAR (4000)

Note About Microsoft Azure SQL Data Warehouse LOB support:

Full LOB data types are not supported. For information on including Limited-size LOB data types in the replication, see the Metadata tab description in Customizing Tasks .

Table 9.24 | Supported Microsoft Azure SQL Data Warehouse Data Types with Mapping from Attunity Replicate Data Types (Cont.)

Attunity Replicate Data Types	Microsoft Azure SQL Data Warehouse Data Types
BLOB	VARBINARY (Max LOB Size * 2) **Note** The maximum LOB size in the Metadata tab cannot exceed 31 KB.
NCLOB	NVARCHAR (Max LOB Size) **Note** The maximum LOB size in the Metadata tab cannot exceed 63 KB.
CLOB	VARCHAR (Max LOB Size) **Note** The maximum LOB size in the Metadata tab cannot exceed 63 KB.

Setting General Connection Properties

This section describes how to configure general connection properties. For an explanation of how to configure advanced connection properties, see **Setting Advanced Connection Properties** below.

To add an Microsoft Azure SQL Data Warehouse Target to Attunity Replicate:

1. In the Attunity Replicate Console, click **Manage Endpoint Connections** to open the **Manage Endpoints Connections** dialog box.
2. In the **Manage Endpoint Connections** dialog box, click **New Endpoint Connection**.
3. In the **Name** field, type a name for your Microsoft Azure SQL Data Warehouse data warehouse [service]. This can be any name that will help to identify your Microsoft Azure SQL Data Warehouse database.
4. In the **Description** field, type a description that helps to identify the Microsoft Azure SQL Data Warehouse target database. This is optional.
5. Select **TARGET** as the **role**.
6. Select **Microsoft Azure SQL Data Warehouse** as the **Type**.
7. Enter the following **Microsoft Azure SQL Data Warehouse** information:
   - **Server name**: Specify the name of the Microsoft Azure SQL Data Warehouse server you are using.
   - **Port**: Specify the port number for Microsoft Azure SQL Data Warehouse.

- **User name**: Specify the user name of a registered Microsoft Azure SQL Data Warehouse user.
- **Password**: Specify the password for the user entered in the **User name** field.
- **Database name**: Specify the target database name.

  If you do not have these values, contact the Microsoft Azure account owner or your company's Microsoft Azure SQL Data Warehouse System Administrator.
- **Azure Blob Storage Access:** During a replication task, Microsoft Azure SQL Data Warehouse authenticates itself to Azure Blob Storage using an SQL Server Credential. You can either configure Replicate to create the Credential automatically during runtime (the default) or use an existing Credential.
  - **Automatically create SQL Server Credential**
  - **Use existing SQL Server Credential**

8. Enter the following Microsoft Azure Blob Storage information. You may need to click the **Microsoft Azure Blob Storage** header to see the information.
   - **Account name**: Specify the name of the Azure Blob Storage account to which you want the files copied.
   - **Container name**: Specify the name of the Azure Blob Storage container to which you want the files copied.
   - **Account key**: Specify the key for your Azure Blob Storage account.
   - **Folder**: Specify the name of the Azure Blob Storage folder to which you want the files copied.

   If you do not have these values, contact the Microsoft Azure Blob Storage account owner or your company's IT department.

   Note
   - This information is case sensitive.
   - To determine if you are connected to the database you want to use or if the connection information you entered is correct, click **Test Connection**.

     If the connection is successful a message in green is displayed. If the connection fails, an error message is displayed at the bottom of the dialog box.

     To view the log entry if the connection fails, click **View Log**. The server log is displayed with the information for the connection failure. Note that this button is not available unless the test connection fails.

Setting Advanced Connection Properties

In the **Advanced** tab, you can set the following properties:

- **Max file size (MB)**: Select or type the maximum size of any CSV file used to transfer data to Microsoft Azure SQL Data Warehouse. The default value is 1024.

- **Number of threads used to upload a file**: Select the number of threads used to upload a single file. The minimum number of threads is 1. The maximum value is 64. The default value is 10.
- **ODBC driver**: The name of the default ODBC driver you are using to connect to Microsoft Azure SQL Data Warehouse. The default driver is **SQL Server Native Client 11.0**.
- **Additional ODBC connection properties**: Enter additional ODBC connection properties if required.

Internal Parameters

Internal parameters are parameters that are not exposed in the UI. You should only use them if instructed by Attunity Support.

To add internal Attunity Replicate parameters:

1. Click the **Internal Parameters** link.

   The **Internal Parameters** dialog box opens.
2. In the edit box, type the name of the parameter you need to add and then click it.
3. The parameter is added to the table below the search box with its default value.
4. Change the default value as required.
5. To reset the parameter value to its default, click the "Restore default value" icon at the end of the row.

Settings Summary

You can view a summary of your settings by clicking the **Setting Summary** link. This is useful if you need to send a summary of your settings to Attunity Support.

Using an IBM Netezza as a Target

This section describes how to set up and use an IBM Netezza database as a target database in a replication task.

In this section:

Prerequisites

Note

- Attunity Replicate must be installed on any Windows computer in your network.
- An IBM Netezza account with the required access privileges is required.

Make sure the following prerequisites have been met:

- IBM Netezza ODBC 64-bit client installed on the Attunity Replicate machine.
- IBM Netezza Tools 7.0.4.2 or above installed on the Attunity Replicate machine. Make sure that the Windows **`Path`** environment variable includes the **`bin`** folder of IBM Netezza Tools (i.e. `installation directory\bin`).

Limitations

Using IBM Netezza as a target database in an Attunity Replicate task is subject to the following limitations:

- The IBM Netezza target database uses the IBM Netezza NZLOAD utility, which does not support loading tables with non-Latin names (e.g. Chinese). If any of your source tables has a non-Latin name, you can map it to a table with a Latin name.

  For more information on mapping table names, see Carrying out General Tasks for a Single Table/View and Defining Global Transformations.
- For IBM Netezza versions before 7.0.3, you need to define a default target schema. For information about defining a target schema, see Target Metadata.

Security Requirements

The Attunity Replicate user must be granted access to the IBM Netezza account as well as the following privileges:

Database Privileges

- LIST on <database> to <ATTUNITY USER>
- SELECT on <database> to <ATTUNITY USER>

Table Privileges

- CREATE TABLE to <ATTUNITY USER>
- LIST on TABLE to <ATTUNITY USER>

Schema Privileges

- CREATE SCHEMA to <ATTUNITY USER>
- LIST on SCHEMA to <ATTUNITY USER>

View Privileges

- SELECT on _T_DATABASE to <ATTUNITY USER>
- SELECT on _V_SCHEMA to <ATTUNITY USER>
- SELECT on _V_USER to <ATTUNITY USER>
- SELECT on _V_TABLE to <ATTUNITY USER>
- SELECT on _V_TABLE_DIST to <ATTUNITY USER>
- SELECT on _V_RELATION_KEYDATA to <ATTUNITY USER>
- LIST on _T_DATABASE to <ATTUNITY USER>
- LIST on _V_SCHEMA to <ATTUNITY USER>
- LIST on _V_USER to <ATTUNITY USER>
- LIST on _V_TABLE to <ATTUNITY USER>
- LIST on _V_TABLE_DIST to <ATTUNITY USER>
- LIST on _V_RELATION_KEYDATA to <ATTUNITY USER>

IBM Netezza Target Data Types

The IBM Netezza database for Attunity Replicate supports most IBM Netezza data types. The following table shows the IBM Netezza target data types that are supported when using Attunity Replicate and the default mapping from Attunity Replicate data types.

Note IBM Netezza does not support applying changes to binary data types in **Batch optimized apply** mode. For more information on **Batch optimized apply** mode, see Changes Processing Tuning.

For information on how to view the data type that is mapped from the source, see the section for the source database you are using.

For additional information about Attunity Replicate data types, see Replicate Data Types.

Table 9.25 | Supported IBM Netezza Data Types with Mapping from Attunity Replicate Data Types

Attunity Replicate Data Types	IBM Netezza Data Types
BOOLEAN	BOOLEAN
BYTES	VARCHAR (Length in Bytes)
DATE	DATE
TIME	TIME
DATETIME	TIMESTAMP
INT1	BYTEINT
INT2	SMALLINT
INT4	INTEGER
INT8	BIGINT
NUMERIC	NUMERIC (p,s)
REAL4	REAL
REAL8	DOUBLE
STRING	VARCHAR (Length)
UINT1	SMALLINT
UINT2	INTEGER
UINT4	BIGINT
UINT8	BIGINT
WSTRING	NVARCHAR (Length)

Note About IBM Netezza LOB support:

Full LOB data types are not supported in the IBM Netezza database. For information on including Limited-size LOB data types in the replication, see the Metadata tab section in Customizing Tasks . Note also that the size of a row in the IBM Netezza database *cannot exceed 64KB*. This should be taken into consideration when specifying the maximum LOB size in the Metadata tab.[second paragraph if needed]

BLOB	VARCHAR (64,000)
NCLOB	NVARCHAR (16,000)
CLOB	VARCHAR (64,000)

Setting General Connection Properties

This section describes how to configure general connection properties. For an explanation of how to configure advanced connection properties, see **Setting Advanced Connection**

Properties below.

To add an IBM Netezza target endpoint to Attunity Replicate:

1. In **Tasks** view, click **Manage Endpoint Connections** to open the **Manage Endpoints Connections** dialog box. Then click the **New Endpoint Connection** button.
2. In the **Name** field, type a name for your database. This can be any name that will help to identify the database being used.
3. In the **Description** field, type a description that helps to identify the IBM Netezza database. This is optional.
4. Select **TARGET** as the database **role**.
5. Select **IBM Netezza** as the database **Type**.
6. In the **Server** field, enter the name of the IBM Netezza server.
7. Optionally, change the default **Port** (5480).
8. Type the IBM Netezza authentication information (**User Name**, **Password**) for the authorized user for this IBM Netezza database. If you do not know this information, see your IBM Netezza database Administrator (DBA).

   Note
   - This information is required. If you are using the **Advanced** tab to create a custom string, make sure to include the **User Name** and **Password** properties. See Setting Advanced Connection Properties for more information.
   - This information is case sensitive.
   - If you want to set custom properties for this database, see Setting Advanced Connection Properties.

   Important: Make sure that the IBM Netezza user entered in the IBM Netezza Authentication section has the correct access privileges. For information on how to provide the required privileges, see Security Requirements.

9. In the **Database name** field, enter the name of the IBM Netezza database.

Setting Advanced Connection Properties

In the **Advanced** tab, you can set the following parameters:

- **Max file size**: Select or type the maximum size (in KB) of a CSV file before the file is loaded into the IBM Netezza database. The default value is 32000 KB.

Internal Parameters

Internal parameters are parameters that are not exposed in the UI. You should only use them if instructed by Attunity Support.

To add internal Attunity Replicate parameters:

1. Click the **Internal Parameters** link.

   The **Internal Parameters** dialog box opens.
2. In the edit box, type the name of the parameter you need to add and then click it.
3. The parameter is added to the table below the search box with its default value.
4. Change the default value as required.
5. To reset the parameter value to its default, click the "Restore default value" icon at the end of the row.

Settings Summary

You can view a summary of your settings by clicking the **Setting Summary** link. This is useful if you need to send a summary of your settings to Attunity Support.

Using MongoDB as a Target

This section describes how to set up and use a MongoDB database as a target endpoint in a replication task.

In this section:

Understanding the Replication Process

This section describes the process by which Replicate adapts the source data to MongoDB conventions. A proper understanding of this process is essential when replicating data to MongoDB.

The following topics are covered:

- How Source Entities are Represented on MongoDB
- Valid Source Structures and Records
- How Replicate Constructs the Target Document and Determines its "_id"
- How Replicate Determines whether to Create or Update Target Documents
- Handling of Source DDL Operations on MongoDB

How Source Entities are Represented on MongoDB

Depending on the source database object type, the source data is replicated to a corresponding database, collection or document on the MongoDB target.

Table 9.26 | Source to MongoDB Object Hierarchy Representation

Object on source database	Object on MongoDB
Relational database: database or Schema File based: Directory/Set of files	database
Relational database:	Collection

Table 9.26 | Source to MongoDB Object Hierarchy Representation (Cont.)

Object on source database	Object on MongoDB
Table » File based: File	
Data record (within a file/table)	JSON Document The JSON will be constructed from the columns that comprise the source data record.

Valid Source Structures and Records

Replicate expects the source records to be in any of the following formats:

- Each record consists of any number of columns.

  In this case the table identifier is expected to be one of the following:

  - A column named "_id" (not necessarily a Primary Key)
  - A Primary Key/Unique Index
  - None of the above
- Each record consists of a single column of a string data type containing valid JSON data (such as may be found on Hadoop). Examples of string data types include `varchar`, `nvarchar`, `text`, `LOB`, and `CLOB`.

  In this case the table identifier is expected to be one of the following:

  - A field named "_id"
  - Nothing

How Replicate Constructs the Target Document and Determines its "_id"

All MongoDB documents must have a document identifier - a unique "_id" field that acts as a primary key. The table below describes how Replicate constructs the target document and determines its "_id".

Data record stored on source as:	Will be stored on MongoDB as:	With the following identifier
One column of a string data type that is identified by a field named "_id".	A JSON document. Replicate assumes that the data in the source column is a valid JSON document and will therefore be replicated to the	"_id" field from the source.

Data record stored on source as:	Will be stored on MongoDB as:	With the following identifier
	target as is.	
One column of a string data type that is NOT identified by a field named "_id".	A JSON document. Replicate assumes that the data in the source column is a valid JSON document and will therefore be replicated to the target as is.	The target document "_id" is not created/updated by the database. Instead: » If the document already exists in the target, then its "_id" remains intact. » If the document does *not* exist in the target, then a new document is created and its "_id" is auto-generated by MongoDB.
Several columns, one of which is named "_id". **Note** The "_id" column does not necessarily have to be Primary Key.	A JSON document constructed from the columns that comprise the source.	"_id" column from the source.
Several columns. The record is identified by a Primary Key or Unique Index (rather than by a column named **"**_id**"**).	A JSON document constructed from the columns that comprise the source.	The identifier depends on the MongoDB database setting in Attunity Replicate. If the **Use source Primary Key or Unique Index as target _id** option is set, then the target document "_id" will comprise the source Primary Key or Unique Index columns. In addition, the Primary Key or Unique Index will also appear as explicit fields in the document. If the **Use source Primary Key or Unique Index as target index** option is set, instead of the target document "_id" being created or updated by the database, one of the following occurs: » If the document already exists on

Data record stored on source as:	Will be stored on MongoDB as:	With the following identifier
		the target, then its "_id" remains intact. » If no such document exists on the target, then a new document is created with an "_id" that has been auto-generated by MongoDB. Also, the source Primary Key or Unique Index segments are used to create a Unique Index on the target.
Several columns without a Primary Key or Unique Index.	A JSON document constructed from the columns that comprise the source.	Instead of the target document "_id" being created or updated by the database, one of the following occurs: » If the document already exists on the target, then its "_id" remains intact. » If no such document exists on the target, then a new document is created with an "_id" that has been auto-generated by MongoDB.

Note

» Fields in the target document that do not exist in the source record will remain untouched.

» When a source record is deleted, the corresponding document will be deleted from the target collection.

How Replicate Determines whether to Create or Update Target Documents

During Change Processing or Full Load to an existing target, Replicate uses the following "matching" logic to determine whether it needs to create a new document on the target or update an existing one.

» If the source record has a column named "_id" (either originally or after global transformations have been applied), then it will be matched to the corresponding "_id" field in the target documents.

Note

- Source column named "_id" refers to a column named "_id" before or after any table or global transformations have been applied.
- If the source record consists of one column of a string data type, then "_id" refers to a field within this column

- If the **Use source Primary Key or Unique Index as target _id** option is set (the default) and the source record does not have a column named "_id", then the source Primary Key or Unique Index segments will be matched to the corresponding "_id" field in the target documents.
- If the **Use source Primary Key or Unique Index as target _id** option is set (the default) and the source record has neither a column named "_id" nor a Primary Key/Unique Index, then all the source columns will be matched to the corresponding fields in the target documents.
- If the **Use source Primary Key or Unique Index as target index** option is set and the source record does not have a column named "_id" but does have a Primary Key/Unique Index, then the source Primary Key/Unique Index segments will be matched to the corresponding Unique Index fields in the target documents.
- **Note:** For proper matching, the index segments in the source table and the target collection should be the same.
- If the **Use source Primary Key or Unique Index as target index** option is set and the following are true:
  - The source record does not have a column named "_id"
  - The source record has a Primary Key/Unique Index
  - The target has no Unique Index
- Then all the source columns will be matched to the corresponding fields in the target documents.
- If the **Use source Primary Key or Unique Index as target index** option is set and the source record has neither a column named "_id" nor a Primary Key/Unique Index, then all the source columns will be matched to the corresponding fields in the target documents.
- If the **Use source Primary Key or Unique Index as target _id** option is set (the default) and the source record does not have a column named "_id", then the source Primary Key or Unique Index segments will be matched to the corresponding "_id" field in the target documents.
- If the **Use source Primary Key or Unique Index as target _id** option is set (the default) and the source record has neither a column named "_id" nor a Primary Key/Unique Index, then all the source columns will be matched to the corresponding fields in the target documents.
- If the **Use source Primary Key or Unique Index as target index** option is set and the source record does not have a column named "_id" but does have a Primary Key/Unique Index, then the source Primary Key/Unique Index segments will be matched

to the corresponding Unique Index fields in the target documents.

Note For proper matching, the index segments in the source table and the target collection should be the same.

- If the **Use source Primary Key or Unique Index as target index** option is set and the following are true:
  - The source record does not have a column named "_id"
  - The source record has a Primary Key/Unique Index
  - The target has no Unique Index
- Then all the source columns will be matched to the corresponding fields in the target documents.
- If the **Use source Primary Key or Unique Index as target index** option is set and the source record has neither a column named "_id" nor a Primary Key/Unique Index, then all the source columns will be matched to the corresponding fields in the target documents.

Note Matching the source columns:
- Match the "Before Image" of the source columns.
- If there is no "Before Image" then it will be assumed that there is not match and the source record will be created as a new document on the target.
- Create/update of the target document is always done using the source's "After Image".

Handling of Source DDL Operations on MongoDB

The following table describes how source DDL operations are handled on MongoDB.

Table 9.27 | DDL Handling

DDL on Source	Behavior on MongoDB
Create table	Create empty collection
Rename table	Rename collection
Truncate table	Configurable according to DDL handling policy: » TRUNCATE target collection » Ignore TRUNCATE
Drop table	Configurable according to DDL handling policy:

Table 9.27 | DDL Handling (Cont.)

DDL on Source	Behavior on MongoDB
	» DROP target collection » Ignore DROP
Add column	» The DDL is ignored and a warning is issued. » When the first INSERT/UPDATE operation is performed on this column, the new field is added to the target document.
Rename column	» The DDL is ignored and a warning is issued. » When the first INSERT/UPDATE operation is performed on this column, the newly named field is added to the target document. » The field with the old name remains as is in the target document (deprecated).
Drop column	» The DDL is ignored and a warning is issued. » The dropped field remains as is in the target document (deprecated).
Change column data type	» The DDL is ignored and a warning is issued. » When the first INSERT operation is performed on this column with the new data type, the target document is created with the field of the new data type.

Change Processing and Error Handling Settings

The following task settings are always used when replicating to a MongoDB target, *regardless of how the settings are actually configured*.

Note If the configured settings differ from the settings listed below, a warning will be issued and the task will continue to run with the allowed settings.

- **Change Processing|Change Processing Tuning|Change processing mode:**
  - Batch optimized apply
  - Allow temporary lapses in transactional integrity to improve performance
- **Change Processing|Apply Changes Settings|When source table is altered:** Ignore ALTER
- **Error Handling|Apply Conflicts|Apply conflict policy:**
  - **No record found for applying DELETE:** Ignore record
  - **Duplicate key when applying an INSERT:** UPDATE the existing target record
  - **No record found for applying an UPDATE:** INSERT the missing target record

Limitations

Using MongoDB as a target database in an Attunity Replicate task is subject to the following limitations:

- Merging multiple source tables into a single MongoDB collection is not supported. In other words, Replicate does not support embedding of source related data via arrays and sub-documents into a single target document.
- Source column names containing a dot character will be replicated in a nested manner instead of as is.
- When processing changes from a source that has LOB columns but does not have Primary Key, the LOB columns will be ignored. From a replication perspective, if the source structure is a single column of a string data type, then its type *should not* be LOB.
- Change Tables: If the source is a table with column named "_id", then the "_id" column of the changed document will appear as "__id" in the Change Table.
- The "No records found for applying DELETE" **Apply Conflicts Policy** is not supported.
- When applying changes from Oracle source to MongoDB target, the Oracle source must be configured with Full Supplemental Logging.

  Note If Full Supplemental Logging is not configured, then columns in the record that were not changed in the source will be loaded into the target with NULL values.

- **MongoDB limitations:** MongoDB collections cannot include the dollar symbol ($) in their name. MongoDB databases cannot include Unicode characters in their name.

MongoDB Target Data Types

The MongoDB database for Attunity Replicate supports most MongoDB data types. The following table shows the MongoDB target data types that are supported when using Attunity Replicate and the default mapping from Attunity Replicate data types.

For information on how to view the data type that is mapped from the source, see the section for the source database you are using.

For additional information about Attunity Replicate data types, see Replicate Data Types.

Table 9.28 | Supported MongoDB Data Types with Mapping from Attunity Replicate Data Types

Attunity Replicate Data Types	MongoDB Data Types
BOOL	BOOLEAN
BYTES	BINARY

Table 9.28 | Supported MongoDB Data Types with Mapping from Attunity Replicate Data Types (Cont.)

Attunity Replicate Data Types	MongoDB Data Types
DATE	DATE
TIME	STRING (UTF8)
DATETIME	DATE
INT1	INT32
INT2	INT32
INT4	INT32
INT8	INT64
NUMERIC	STRING (UTF8)
REAL4	DOUBLE
REAL8	DOUBLE
STRING	If the column is recognized as a JSON, then load it to the target as a document. Otherwise, map to STRING (UTF8).
UINT1	INT32
UINT2	INT32
UINT4	INT64
UINT8	STRING (UTF8)
WSTRING	If the column is recognized as a JSON, then load it to the target as a document. Otherwise, map to STRING (UTF8).
BLOB	BINARY
NCLOB	If the column is recognized as a JSON, then load it to the target as a document. Otherwise, map to STRING (UTF8).
CLOB	If the column is recognized as a JSON, then load it to the target as a document. Otherwise, map to STRING (UTF8).

Controlling the Target Structure

You can control the target structure by renaming source columns and tables either manually before the task begins or in a task transformation. Several renaming options are available, depending on the desired target structure. These are described below.

- You can prefix the source column name with `$JSON`: (i.e. `$JSON:column_name`) either manually or using a transformation. In this case, the column will be created as a nested JSON document within the target document (rather than as a STRING field)."

Note In order to be considered as a JSON type, in addition to the requisite naming convention described above, the column data must also be in valid JSON format.

Example:

Original Source Column Name	Renamed Source Column
ContactDetails	$JSON:ContactDetails

The `ContactDetails` column is replicated as a nested JSON within each document (rather than as a string field):

```
{
  "_id" : "1",
  "FirstName‟ : "David",
  "LastName‟ : "Kimbel",
   "ContactDetails" :
  {
    "Home" :
    {
       "Address" : "Boston",
       "Phone" : "1111111111"
    },
    "Work" :
    {
       "Address" : "Boston",
       "Phone" : "2222222222"
    }
  }
}
```

Without column renaming, this would have been the result:

```
{
   "_id" : "1",
   "FirstName” : "David",
   "LastName" : "Kimbel",
   "ContactDetails" : "{"Home" : { "Address" : "Boston", "Phone" :
  "1111111" },
   "Work" : { "Address" : "Boston", "Phone" : "2222222222" } }"
}
```

» You can rename a column so that it consist of several parts separated with a dot

character either manually or using a transformation. In this case, the "dotted" columns will be created as a nested JSON document within the target document.

Example:

In the following example, a transformation has been used to rename the source columns as follows:

Original Source Column Name	Renamed Source Column
HomeAddress	ContactDetails.Home.Address
HomePhone	ContactDetails.Home.Phone
WorkAddress	ContactDetails.Work.Address
WorkPhone	ContactDetails.Work.Phone

This creates the following JSON doc on the target:

```
{
   "_id" : "1",
   "FirstName" : "David",
  "LastName" : "Kimbel",
  "ContactDetails" :
  {
    "Home" :
    {
       "Address" : "Boston",
       "Phone" : "1111111111"
    },
    "Work" :
    {
      "Address" : "Boston",
      "Phone" : "2222222222"
    }
  }
}
```

Without column renaming, this would have been the result:

```
{
   "_id" : "1",
   "FirstName" : "David",
   "LastName" : "Kimbel",
   "HomeAddress" : "Boston",
   "HomePhone" : "1111111111",
```

```
   "WorkAddress" : "Boston",
   "WorkPhone" : "2222222222"
}
```

- You can prefix a column name with `$ARRAY:` (i.e. `$ARRAY:column_name`), either manually or using a transformation. In this case, the column will be considered as an `ARRAY` type and will be created as such in the target document.

  Note In order to be considered as an `ARRAY` type, in addition to the requisite naming convention described above, the column data must also be in valid JSON Array format (e.g. `["elem1" , "elem2", ...]`).

  Example:

Original Source Column Name	Renamed Source Column
ContactAddress	$ARRAY:ContactAddress
ContactPhoneNumbers	$ARRAY:ContactPhoneNumbers

  The columns will be created as repeated elements (ARRAY) within the target document rather than as STRING fields:

```
{
   "_id" : "1",
   "FirstName‟ : "David",
   "LastName‟ : "Kimbel",
   "ContactAddresses" : [
    "Boston" ,
    "New York"
   ],
   "ContactPhoneNumbers" : [
    "1111111111" ,
    "2222222222"
   ]
}
```

  Without column renaming, this would have been the result:

```
{
   “_id” : “1”,
   “FirstName” : “David",
  “LastName” : “Kimbel",
    “ContactAddresses” : “["Boston", “New York”]”,
     “ContactPhoneNumbers” : “["1111111111“, "2222222222“]”
```

```
}
```

Setting General Connection Properties

This section describes how to configure general connection properties. For an explanation of how to configure advanced connection properties, see **Setting Advanced Connection Properties** below.

For more information on adding an endpoint to Attunity Replicate, see Working with Endpoints.

To add a MongoDB target endpoint to Attunity Replicate:

1. In the Attunity Replicate console, click **Manage Endpoint Connections** to open the **Manage Endpoint Connections** dialog box and then click **New Endpoint Connection**.
2. In the **Name** field, type a name for your database. This can be any name that will help to identify the database being used.
3. In the **Description** field, type a description that helps to identify the MongoDB database. This is optional.
4. Select **TARGET** as the database **role**.
5. Select **MongoDB** as the database **Type**.
6. In the **Server** field, specify one or more MongoDB server hosts using the following format (for high availability):

   `host1[:port1][,host2[:port2]]`

   Example:

   `192.168.1.100:27017,192.168.1.101:1234`

   Replicate will connect to the first available host. If a host is specified without a port then port 27017 will be used as the default.
7. Choose either **Password** or **None** as the authentication **Type**.
8. If you chose **Password** as the authentication type, specify the **User name** and **Password** of a valid user on the target MongoDB database. The user should also have the necessary permissions to access the specified database.

   Note If **Password** is set as the authentication "Type", the following prerequisites apply:

   - The specified User-Password must exist in the authentication database and be a valid combination on the MongoDB cluster.
   - The specified user must be granted the `readWrite` role on the target database(s).
   - To list endpoints, the specified user must be granted with a role that allows the `listDatabases` action. This action is included, for example, in the following roles: `clusterMonitor`, backup and `readAnyDatabase`
9. If you chose **Password** as the authentication type, choose an authentication

Mechanism from the drop-down list.

Note When "Default" is selected, Replicate will use SCRAM-SHA-1 for MongoDB3.x and MONGODB-CR for MongoDB2.x.

Note The **x.509 Certificate** and **Kerberos** authentication protocols are not supported.

10. If you chose **Password** as the authentication type and *did not* choose `PLAIN (LDAP SASL)` as the **Mechanism**, either specify the name of your **Authentication database** or leave the default ("admin").

    When adding a user to MongoDB, you create the user in a specific database. This database is the *authentication database* for that user. Together, the user's name and database serve as a unique identifier for that user.
11. To establish a secure connection between Replicate and MongoDB, select the **Use SSL** check box.
12. Expand the **Target** section and then choose on the following **Load the source schemas into** options:
    - **The following database** - Select this option if you want all source schemas to be loaded into a single database.

      Note When this option is selected, the following occurs:
      - If a **Target table schema** is defined, it will be ignored.
      - If neither a **Target control schema** nor a **Target table schema** are defined, the task Control Tables will be created in the specified target database.

    - **Multiple endpoints (one for each schema)** - Select this option if you want each source schema to be loaded into a corresponding target database.

      Note When this option is selected, the following occurs:
      - Replicate will create the corresponding endpoints if they do not already exist on the target.
      - If neither a **Target control schema** nor a **Target table schema** are defined, the task Control Tables will be created in a database named `attrep_control`. In such a case, the specified user must be granted the `readWrite` role for the `attrep_control` database.
      - For to ARC-based sources, the **Target table schema** will be used as the target database for all the source tables. If the **Target table schema** is not set, then a default database named **attrep_control** will be used as the target database for all the source tables.

For information on defining target table schemas and target control schemas, see Target Metadata.

Setting Advanced Connection Properties

In the **Advanced** tab, you can determine how Replicate should behave if the source does not have a column named "_id". If the source *does* have a column named "_id", then it will be used to identify the documents in the target collection.

If the source does not have a column named "_id", the following options are available:

- **Use source primary key or unique index as target _id**: When this option is selected, the following occurs:
  - The source Primary Key columns (or the Unique Index columns in the absence of a Primary Key) will form the target `_id`.
  - The source Primary Key columns (or the Unique Index columns in the absence of a Primary Key) will appear in the replicated target document both as an identifier (`_id`) and as explicit fields in the document.

    Selecting this option preserves the Primary Key from the source, facilitating both identification of the target documents and direct access to them.
- **Use source primary key or unique index as target index**: When this option is selected, the following occurs:
  - The source Primary Key columns (or the Unique Index columns in the absence of a Primary Key) will not form the target document "_id". Instead, the "_id" will either remain intact for existing target documents or auto-generated by MongoDB for new target documents.

    Note To facilitate comparison between the source Primary Key or Unique Index fields and the target Unique Index fields, both the source table index and the target collection index must consist of the same segments.

  - An index formed from the source Primary Key columns (or the Unique Index columns in the absence a Primary Key) will be created in the target collection.

    Benefits of this option:

    This option offers the following benefits:

    - In MongoDB, document "_id" is immutable. If the source Primary Key/Unique Index columns are subject to change, then they should not be used to create the target "_id". However, using them to create a target index will facilitate rapid access.
    - Prevents the Primary Key/Unique Index from appearing both within the "_id" and as explicit fields within the document (current behavior). This can be useful if the Primary Key/Unique Index is very large.

For further information about these options, see:

- Understanding the Replication Process
- How Replicate Constructs the Target Document and Determines its "_id"

» How Replicate Determines whether to Create or Update Target Documents

Internal Parameters

Internal parameters are parameters that are not exposed in the UI. You should only use them if instructed by Attunity Support.

To add internal Attunity Replicate parameters:

1. Click the **Internal Parameters** link.

   The **Internal Parameters** dialog box opens.
2. In the edit box, type the name of the parameter you need to add and then click it.
3. The parameter is added to the table below the search box with its default value.
4. Change the default value as required.
5. To reset the parameter value to its default, click the "Restore default value" icon at the end of the row.

Settings Summary

You can view a summary of your settings by clicking the **Setting Summary** link. This is useful if you need to send a summary of your settings to Attunity Support.

Using Kafka as a Target

This section describes how to set up and use Kafka as a target endpoint in a replication task. In a task with a Kafka target endpoint, each source record is transformed into a message which is then written (with an optional message key) to a partition in the specified topic.

In this section:

Transaction Processing by the Consumer

When configuring the Attunity Replicate Kafka endpoint, users can configure various settings that affect where messages are published within the Kafka infrastructures (topics/partitions).

During a task's CDC stage, committed changes that are detected by the Attunity Replicate source endpoint are grouped by transaction, sorted internally in chronological order, and then propagated to the target endpoint. The target endpoint can handle the changes in various ways such as applying them to the target tables or storing them in dedicated Change Tables.

Each CDC message has both a transaction ID as well as change sequence. As the change sequence is a monotonically growing number, sorting events by change sequence always achieves chronological order. Grouping the sorted events by transaction ID then results in transactions containing chronologically sorted changes.

However, as Kafka is a messaging infrastructure, applying changes is not feasible while storing changes in tables is meaningless. The Replicate Kafka endpoint, therefore, takes a different approach, which is to report all transactional events as messages.

How it Works

Each change in the source system is translated to a data message containing the details of the change including the transaction ID and change sequence in the source. The data message also includes the changed columns before and after the change. As explained

above, the order in which the Kafka target writes the messages is the same as order of changes within each transaction.

Once a data message is ready to be sent to Kafka, the topic and partition it should go to are determined by analyzing the endpoint settings as well as potentially transformation settings. For example, the user might decide to configure the endpoint in such a way that every table is sent to a different topic and set the partition strategy to "Random", meaning that each message (within the same table) will be sent to a different partition.

Transaction Consistency from a Consumer Perspective

If maintaining transaction consistency is important for the consumer implementation, it means that although the transaction ID exists in all data messages, the challenge is to gather the messages in a way that would facilitate identifying a whole transaction. An additional challenge is getting the transaction in the original order they were committed, which could be an even greater challenge if transactions are spread across multiple topics and partitions.

The simplest way of achieving the above goal is to direct Replicate to a specific topic and a specific partition (in the endpoint settings). This means that all data messages will end up in a single partition, thus guaranteeing ordered delivery both of transactions *and* of changes within a transaction. The consuming application could then consume messages - accumulating a transaction in some intermediate memory buffer - and when a new transaction ID is detected, mark the previous transaction as completed.

Although the simple way may work, it's not very efficient at the task level as all messages end up in the same topic and partition, not necessarily utilizing the full parallelism of the Kafka cluster. This may be a non-issue if there are multiple tasks, each taking advantage of a different topic/partition. In such as scenario, the gathering of messages from those tasks may very well utilize the cluster optimally.

The more generic way where data may be spread over multiple topics and partitions means that some intermediate buffer such as memory, a table in a relational database, or even other Kafka topics would need to be used to collect information about transactions. Then, the transactions would need to be rebuilt by periodically (every few minutes/hours) sorting the events collected from Replicate's Kafka output by the change sequence and grouping them by transaction ID.

Limitations

When defining a task with Kafka as the target endpoint, the following limitations apply:

- When replicating from an Oracle source, full supplemental logging must be enabled for all of the source tables.
- The Kafka target endpoint does not support unlimited LOB size. Therefore, when replicating from source tables with LOB columns, *do not* select the **Allow unlimited LOB size** option.

  For more information on defining LOB settings, see Target Metadata.
- **Batch optimized apply** mode is not supported. If this mode is set, the task will

automatically switch to **Transactional apply** mode and issue an appropriate warning.

For more information on these modes, see Changes Processing Tuning.

- Store Changes mode is not supported.

  For more information on Store Changes mode, see Setting up Tasks.

- Kafka topic names cannot exceed 255 characters (249 from Kafka 0.10) and can only contain the following characters:

  a-z|A-Z|0-9|. (dot)|_(underscore)|-(minus)

  If the source table names exceed the maximum permitted length or contain unsupported characters, you need to either modify the names before starting the task or define a global transformation. For information on defining global transformations, see Defining Global Transformations.

Kafka Target Data Types

The following table shows the default mapping from Attunity Replicate data types to Kafka data types.

For information on source data type mappings, see the section for the source endpoint you are using.

For additional information about Attunity Replicate data types, see Replicate Data Types.

Note When using the JSON message format, binary values are represented as hexadecimal digits.

Table 9.29 | Supported Kafka Target Data Types with Mapping from Attunity Replicate Data Types

Attunity Replicate Data Types	Kafka Target Data Types
DATE	DATE
TIME	TIME
DATETIME	DATETIME
BYTES	BYTES (length)
BLOB	BLOB
REAL4	REAL4 (7)
REAL8	REAL8 (14)
INT1	INT1 (3)
INT2	INT2 (5)
INT4	INT4 (10)

Table 9.29 | Supported Kafka Target Data Types with Mapping from Attunity Replicate Data Types (Cont.)

Attunity Replicate Data Types	Kafka Target Data Types
INT8	INT8 (19)
UINT1	UINT1 (3)
UINT2	UINT2 (5)
UINT4	UINT4 (10)
UINT8	UINT8 (20)
NUMERIC	NUMERIC (p,s)
STRING	STRING (Length)
WSTRING	STRING (Length)
CLOB	CLOB
NCLOB	NCLOB
BOOLEAN	BOOLEAN (1)

Setting General Connection Properties

This section describes how to configure general connection properties. For an explanation of how to configure advanced connection properties, see **Setting Advanced Connection Properties** below.

To define the general connection properties:

1. Click the **Manage Endpoint Connections** toolbar button.
   The **Manage Endpoints Connections** dialog box opens.
2. Click the **New Endpoint Connection** toolbar button.
   The **Name**, **Description**, **Type** and **Role** fields are displayed on the right.
3. In the **Name** field, specify a display name for the endpoint.
4. In the **Description** field, optionally type a description for the Kafka endpoint.
5. Select **Target** as the endpoint **Role**.
6. Select **Kafka** as the endpoint **Type**.
   The dialog box is divided into **General** and **Advanced** tabs.
7. In the **Broker servers** field, specify one or more broker servers using the following format (for high availability):
   `server1[:port1][,server2[:port2]]`
   Example:
   `192.168.1.100:9092,192.168.1.101:9093`

Replicate will connect to the first available host. If a host is specified without a port then port 9092 will be used as the default.

Note All of the broker servers in your cluster need to be accessible to Replicate. However, you do not need to specify all of the servers in the **Broker servers** field. This is because Replicate only need to connect to one of the servers in order to retrieve the connection details for the other servers in the cluster. It is therefore best practice to specify the servers that are most likely to be available when the task is run. The servers to which Replicate produces messages is determined by the topic and partitioning topic and partitioning settings described below.

8. In the **Security** section, set the following properties:

Note

- The **Use SSL** and **Certificate** authentication options are only supported from Kafka 0.9 and above.
- The CA file, public key file and private key file must all be in PEM format.
- The **Kerberos** and **User name and password** authentication methods are only supported from Kafka 0.10 and above.
- All of the broker servers in the cluster must be configured to accept connection requests using the selected **Authentication** method.

- **Use SSL** (supports TLS 1.0, 1.1 and 1.2): Select this option to encrypt the data between the Replicate machine and the broker server(s). If the brokers are configured to require SSL, then you *must* select this option.
  - **CA path**: Specify the directory containing the CA (Certificate Authority) certificate or the full path to a specific CA certificate.
- **Authentication**: Select one of the following:
  - **None** - To send messages without authentication.
  - **Certificate** - If you select this option, you also need to provide the following information:
    - **Public key file** - The full path to the public key file on the Replicate Server machine.
    - **Private key file** - The full path to the private key file on the Replicate Server machine.
    - **Private key password** - The password for the private key file.
  - **Kerberos** - Currently this option is only supported when Replicate Server is installed on a Linux machine. If you select this option, you also need to provide the following information:
    - **Principal** - The Kerberos principal used to authenticate against the broker server(s).

- **Keytab file** - The full path to the keytab file (that contains the specified principal) on the Replicate Server machine.

- **User name and password** - Currently this option is only supported when Replicate Server is installed on a Linux machine. You can select this option to authenticate yourself using a user name and password (SASL/PLAIN). To prevent the password from being sent in clear text, it is strongly recommended to enable the Use SSL option as well.

9. In the **Data Publishing** section, set the following properties:
   a. In the **Publish the data to** field, choose one of the following:
      - **Specific topic** - to publish the data to a single topic. Either type a topic name or use the browse button to select the desired topic.
      - **Specific topic for each table** - to publish the data to multiple topics corresponding to the source table names.

      Note If the topics do not exist, configure the brokers with `auto.create.topics.enable=true` to enable Replicate to create the topics during runtime. Otherwise, the task will fail.

   b. From the **Partition strategy** drop-down list, field, select either **Random** or **By message key**. If you select **Random**, each message will be written to a randomly selected partition. If you select **By message key**, messages will be written to partitions based on the selected **By message key** (described below).
   c. From the **Message key** drop-down list, field, select one of the following:
      - **None** - To create messages without a message key.
      - **Schema and table name** - For each message, the message key will contain a combination of schema and table name (e.g. "`dbo+Employees`").

        When **By message key** is selected as the **Partition strategy**, messages consisting of the same schema and table name will be written to the same partition.
      - **Primary key columns** - For each message, the message key will contain the value of the primary key column.

        When **By message key** is selected as the **Partition strategy**, messages consisting of the same primary key value will be written to the same partition.
10. In the **Message Properties** section, set the following properties:
    a. Choose **JSON** or **Avro** as the message format.
    b. To publish the schema message (for the corresponding data message) to a topic, select the **Publish schema to topic** check box and then either type the topic name or use the Browse button to select the desired topic. This option is required if the message format is set to **Avro** since Avro-formatted messages can only be opened using the Avro schema.

Note Attunity provides an Avro Message Decoder SDK for consuming Avro messages produced by Attunity Replicate. You can download the SDK together with the Avro Message Decoder Developer's Guide as a ZIP file from the Customer Zone.

An understanding of the Attunity envelope schema is a prerequisite for consuming Avro messages produced by Attunity Replicate. If you do not wish to use the SDK, see The Attunity Envelope for a description of the Attunity envelope schema.

Note It is strongly recommended *not* to publish schema messages to the same topic as data messages.

Note If the topics do not exist, configure the brokers with `auto.create.topics.enable=true` to enable Replicate to create the topics during runtime. Otherwise, the task will fail.

c. From the **Compression** drop-down list, optionally select one of the available compression methods (**Snappy** or **gzip**). The default is **None**.

Overriding the Default Settings

A transformation can be defined that overrides the topic, partition and message key settings defined in the **General** tab.

Note Before you can define such a transformation, you first need to add a source endpoint to the task and select the tables you want to replicate.

To define a transformation:

1. Open the task you defined.
2. If you are defining a transformation for a single table, select one of the source tables. Otherwise, skip to Step 3.
3. Define a transformation that adds one of the following columns:
   - `$topic` - To write messages to a specific topic.
   - `$partition` - To write messages to a specific partition.
   - `$key` - To create a custom message key.

   For information on creating a transformation for a single table, see Defining Transformations for a Single Table/View.

   For information on creating a global transformation rule, see Defining Global Transformations.

4. Define an expression for the new column that returns the following values:
   - For a `$topic` column, the expression should return the topic name.
   - For a `$partition` column, the expression should return the partition number. Note that an error will be returned during runtime if the partition number does not exist.
   - For a `$key` column, the expression should return the message key contents.

   For information on creating expressions, see Using the Expression Builder (for Filters, Transformations, and Global Transformations).

Setting Advanced Connection Properties

In the **Advanced** tab, you can define advanced properties for the Kafka target endpoint:

- **Message Maximum Size**

  In the **Message maximum size** field, specify the maximum size of messages that the broker(s) are configured to receive (`message.max.bytes`). Replicate will not send messages larger than the maximum size.

Internal Parameters

Internal parameters are parameters that are not exposed in the UI. You should only use them if instructed by Attunity Support.

To add internal Attunity Replicate parameters:

1. Click the **Internal Parameters** link.

   The **Internal Parameters** dialog box opens.
2. In the edit box, type the name of the parameter you need to add and then click it.
3. The parameter is added to the table below the search box with its default value.
4. Change the default value as required.
5. To reset the parameter value to its default, click the "Restore default value" icon at the end of the row.

Settings Summary

You can view a summary of your settings by clicking the **Setting Summary** link. This is useful if you need to send a summary of your settings to Attunity Support.

The Attunity Envelope

All Attunity message types covered in this section are encapsulated in a single message schema called the Attunity Envelope. The schema of the Attunity envelope is as following:

```
{
    "type":"record",
    "name":"MessageEnvelope",
    "fields":[
          {"name":"magic","type":{"type":"fixed","name":"Magic","size":5}},
```

```
        {"name":"type","type":"string"},
        {"name":"headers","type":["null",{"type":"map","values":"string"}]},
        {"name":"messageSchemaId","type":["null","string"]},
        {"name":"messageSchema","type":["null","string"]},
        {"name":"message","type":"bytes"}
    ]
}
```

The fields in the envelope are as follows:

- `magic` (5 bytes fixed field)

  The constant "atMSG" is used to identify this form of message. The "atMSG" constant should be used to validate that this message is indeed an Attunity envelope message.

- `type` (string field)

  Describes the enveloped message type. This can be one of two values: `MD` which stands for metadata message and `DT` which stands for data message.

- `headers` (map of string key and value)

  A free for use map for various properties set at the application level. Currently, no headers are set by Attunity Replicate but this may change in future versions.

- `messageSchemaId` (null or string)

  A reference to a schema defined elsewhere, which can be used to deserialize the bytes in the message field. This specification does not explain how the schema ID is used for looking up the actual schema - it is an application level detail. This field is used exclusively with the `messageSchema` field.

- `messageSchema` (null or string)

  An embedded UTF-8 encoded Avro JSON schema with which the message field can be serialized. This field is used exclusively with the `messageSchemaId` field.

- `message` (bytes)

  An Avro encoded message, which is the payload of the message envelope.

Given the envelope schema, it is possible for anyone using this schema to properly decode the envelope messages from Kafka.

Once the envelope message has been decoded, there are two possible scenarios:

- **Scenario 1:** Decoding a self-describing message such as the metadata message
- **Scenario 2:** Decoding a message by referenced schema ID such as data messages

The method for logically decoding messages in both scenarios is described below.

Decoding a Self-Describing Message

When the `messageSchema` field is not null, it means the `message` field can be decoded using the schema included in the `messageSchema` field. This is fairly straightforward to perform programatically since the only thing you need to usually supply Avro is a schema and a message, both of which are provided in the envelope message.

The Attunity metadata messages which include both table metadata, lineage and data schema description (to be referenced later by data messages) are enveloped in the self-describing envelope.

Decoding a Message by Referenced Schema ID

Avro schemas are JSON documents which can be quite large, usually much larger than the data encoded by Avro conforming to the schema. For example, a schema of a 10 column table could be a JSON document of more than 100 characters while an actual row encoding of 10 columns may be only 10 bytes (depending of course on the type and length of fields). It is therefore typically not recommended to include schema and data together in a Kafka message because the schema information is redundant and is the same for all data messages while the actual data is the only thing which differs between data messages.

To avoid sending schema with each data message, each schema has a 32 bytes long ID. When a data message based on a previously sent data message schema (via the metadata message) is constructed, the `messageSchema` field is set to null and the `messageSchemaId` field is set to the 32 bytes ID of the schema instead. The application responsibility is to locate the data schema sent earlier in the metadata message and use that schema to decode the data message contained in the `message` field.

Typical Consumer Logic

A typical scenario involving Kafka involves Attunity Replicate as the Producer of messages into Kafka and customer code as the Consumer. Attunity Replicate offers the ability to define a specific topic as the schema topic and different topics for the table data.

The customer's consumer code should read metadata messages from the schema topic and then save the data schemas and any other information the consumer wishes to access later in a customer defined zone. Another set of customer consumers should read data messages from the various data topics, and access the data schemas zone as required to retrieve the data schemas required for decoding the data messages.

When consuming data messages and metadata messages from several topics and partitions in a multi-thread/process manner, a situation may arise where a given consumer may attempt to read a data message *before* the corresponding metadata message has been read. As it is not possible to read a data message before its corresponding metadata message, the consumer's logic should wait a reasonable amount of time until the corresponding metadata message has been read. If the metadata message is still not available after waiting for a reasonable amount of time, the consumer should handle this as an unexpected error and activate the planned error policy. An example of such a policy could be saving the message in a dedicated “delayed” topic for later processing.

As a rule of thumb, the number of metadata messages will be much lower (in the magnitude of 1:10000 or more) than the number of data messages. So, assuming a metadata consumer is active, the gap between metadata message and data message should be no more than a few seconds (usually, milliseconds).

Metadata and Data Messages

This topic describes the structure and content of the Metadata and Data messages produced by the Replicate Kafka endpoint.

Metadata Message

Field	Type	Description
schemaId	String	The unique identifier of the Avro schema.
lineage	Structure	Information about the origin of the data (Replicate server, task, table, and so on)
server	String	The name of the Replicate server.
task	String	The name of the task.
schema	String	The name of the database schema.
table	String	The name of the table.
tableVersion	Integer	Replicate maintains a version number of the structure of source table. Upon DDL change on the source, the version is increased and a new metadata message is produced.
timestamp	String	The date and time of the metadata message.
tableStructure	Structure	Describes the structure of the table.
tableColumns	Structure	Contains the list of columns and their properties.
{columns}	Structure	For each column, a record with the below properties.
ordinal	Integer	The position of the column in the record.
type	String	The column data type.
length	Integer	The maximum size of the data (in bytes) permitted for the column.
precision	Integer	For NUMERIC data type, the maximum number of digits required to represent the value.
scale	Integer	For NUMERIC data type, the maximum number of digits to the right of the decimal point permitted for a number.
primaryKeyPosition	Integer	The position of the column in the table's Primary Key. or Unique Index. The value is zero if the column is not part of the table's Primary Key.
dataSchema	String	The Avro schema for deserializing the Data messages.

Data Message

Field	Type	Description
headers	Structure	Information about the current record
operation	Enum	The operation type. Full Load (Replicate transfers the existing records from source table) REFRESH – insert of a record during Full Load stage. CDC (Replicate transfers the changes from source table) INSERT – insertion of new record UPDATE – update of existing record DELETE – deletion of a record
changeSequence	String	A monotonically increasing change sequencer that is common to all change tables of a task. Use this field to order the records in chronological order. Applicable to CDC operations.
timestamp	String	The original change UTC timestamp. Applicable to CDC operations.
streamPosition	String	The source CDC stream position. Applicable to CDC operations.
transactionId	String	The ID of the transaction that the change record belongs to. Use this field to gather all changes of a specific transaction. Applicable to CDC operations.
data	Structure	The data of the table record
{columns}		The column names and values in the current record.
beforeData	Structure	The data of the table record, before the change
{columns}		The column names and values, before the change. Applicable to UPDATE operation.

Using Teradata Aster as a Target

This section describes how to set up and use Teradata Aster as a target endpoint in a replication task.

In this section:

Prerequisites

Note

- Attunity Replicate must be installed on any Windows computer in your network.
- A Teradata Aster account with the required access privileges is required.

Make sure the following prerequisites have been met:

- Teradata Aster ODBC Driver version 5.11 installed on the computer where Attunity Replicate is located.

Security Requirements

You must provide Teradata Aster account access to the Attunity Replicate user. This user must have read/write privileges in the Teradata Aster database.

Teradata Aster Target Data Types

The Teradata Aster endpoint for Attunity Replicate supports most Teradata Aster data types. The following table shows the Teradata Aster target data types that are supported when using Attunity Replicate and the default mapping from Attunity Replicate data types.

For information on how to view the data type that is mapped from the source, see the section for the source endpoint you are using.

For additional information about Attunity Replicate data types, see Replicate Data Types.

Table 9.30 | Supported Teradata Aster Data Types with Mapping from Attunity Replicate Data Types

Attunity Replicate Data Types	Teradata Aster Data Types
BOOLEAN	BOOLEAN
BYTES	BYTEA
DATE	DATE
TIME	TIME (s)
DATETIME	TIMESTAMP (s)
INT1	SMALLINT
INT2	SMALLINT
INT4	INTEGER
INT8	BIGINT
NUMERIC	NUMERIC (p,s)
REAL4	REAL
REAL8	FLOAT8
STRING	VARCHAR (Length)
UINT1	SMALLINT
UINT2	INTEGER
UINT4	BIGINT
UINT8	BIGINT
WSTRING	VARCHAR (Length)
BLOB	BYTEA
NCLOB	TEXT
CLOB	TEXT

Setting General Connection Properties

You can add Teradata Aster to Attunity Replicate to use as a target. For information on how to add endpoints, see Working with Endpoints.

To add a Teradata Aster target endpoint to Attunity Replicate:

1. In the Attunity Replicate console, click **Manage Endpoint Connections** to open the **Manage Endpoints Connections** dialog box. Then click the **New Endpoint Connection** toolbar button.
2. In the **Name** field, type a name for your endpoint. This can be any name that will help to identify the endpoint being used.

3. In the **Description** field, type a description that helps to identify the Teradata Aster end-point. This is optional.
4. Select **Target** as the endpoint **role**.
5. Select **Teradata Aster** as the endpoint **Type**.
6. In the **Server** field, enter the name of the Teradata Aster server.
7. Optionally, change the default **Port** (5433).
8. Type the Teradata Aster authentication information (**User Name**, **Password**) for the authorized user for this Teradata Aster database. If you do not know this information, see your Teradata Aster database Administrator (DBA).

   Note This information is case sensitive.

   Important: Make sure that the Teradata Aster user entered in the Teradata Aster Authentication section has the correct access privileges. For information on how to provide the required privileges, see Security Requirements.

9. In the **Database name** field, enter the name of the Teradata Aster database.

Setting Advanced Connection Properties

In the **Advanced** tab, you can set the following parameters:

- **Max file size**: Select or type the maximum size (in KB) of a CSV file before the file is loaded into the Teradata Aster database. The default value is 32000 KB.

Internal Parameters

Internal parameters are parameters that are not exposed in the UI. You should only use them if instructed by Attunity Support.

To add internal Attunity Replicate parameters:

1. Click the **Internal Parameters** link.

   The **Internal Parameters** dialog box opens.
2. In the edit box, type the name of the parameter you need to add and then click it.
3. The parameter is added to the table below the search box with its default value.
4. Change the default value as required.
5. To reset the parameter value to its default, click the "Restore default value" icon at the end of the row.

Settings Summary

You can view a summary of your settings by clicking the **Setting Summary** link. This is useful if you need to send a summary of your settings to Attunity Support.

10 | Using the Attunity Replicate File Channel

This section describes how to use the Attunity Replicate file channel as a source or target in a replication task.

In this chapter:

Setting Up Attunity Replicate File Channel Tasks

To replicate data using the file channel, you must set up two tasks of the following type:

- Local Task
- Remote Task

Note When using file channel, Change Tables can be enabled for the remote task *but not for the local task* (enabling Change Tables for the local task will result in remote task failure).

Local Task

You set up the local task using the File-Channel endpoint as a target. The binary file created in this task is used as the source for one or more remote tasks using the File-Channel source endpoint.

The local task replicates data from an Attunity Replicate supported endpoint to the file channel. If you changed the default folder for storing data files (during the installation), then you must specify the location of the binary file created by the file channel. This location can be anywhere in your system. For more information on setting up a local task, see Using the File Channel as a Target.

Remote Task

Remote tasks use the File Channel as a source endpoint. You use the file created by the local task for this source. You can replicate the data to any endpoint that is supported by Attunity Replicate. You define the location of the File-Channel file as the remote location where the file was created. The data is pushed over the network to the defined location anywhere in your system. You can also define more than one location for the replicated data. In this case, define a separate remote task for each location.

If you want to push the data to an endpoint that is not in your LAN, use the File Transfer Service to send the files created in the local task to the remote location.

When you run the remote task, data is sent to the target in the following instances:

- The first time you run the task as a full load.
- Each time changes are made to the file. In this case, change processing takes place.

When the remote task runs, it will continuously look for the source file until the task is stopped. When the file is found, the data is replicated to the target endpoint. If no source file is found, an error is displayed; however, the task will continue to check for the correct file. Therefore, it is recommended that you run the local task first to ensure that the file exists.

Note To replicate tables that were added to the local file channel task after the initial full load, you need to reload both the local and the remote file channel tasks.

For more information on setting up a remote task, see Using the File Channel as a Source.

Replicating to Multiple Targets (Distribution)

You can use the File Channel to distribute from a single source endpoint to multiple targets, either of the same type (e.g. Microsoft SQL Server to Microsoft SQL Server) or of different types (e.g. Microsoft SQL Server to Oracle and SAP Sybase ASE).

To do this:

1. For each of the target endpoints, define a separate (remote) task that replicates from the File Channel source to the target endpoint. In the **Advanced** tab of the File Channel source settings, make sure to clear the **Delete processed files** check box. This ensures that the File Channel files will be available for distribution as required.
2. Define a local task that replicates from the source endpoint to a File Channel target.
3. Run the local task (this will create the File Channel files required by the remote task).
4. For each of the remote tasks, select which tables to replicate (from the File Channel source) and optionally apply Filters and Transformations to them.
5. Run the remote tasks.

For more information on defining tasks, see Designing Tasks.

For information on Filters and Transformations, see Customizing Tasks .

Note By default, all the metadata for the selected source tables is replicated from the Local Task to the Remote Task. This allows you to remove, add and transform tables in the remote task as needed. However, if you want the tables in the source and target endpoints to be identical, you can prevent replication of the metadata (and thereby shorten the processing time) by specifying `provideremotemetadata=N` in the **Override connection string parameters** field of the File Channel target's **Advanced** tab.

Adding Tables to a Running Remote Task

When distributing to multiple targets, it is possible to replicate a different subset of tables to each target if necessary. Before starting the task, you can select which tables to replicate using the standard procedure described in Adding Tables and/or Views to a Task. However, if the task is already running, you need to perform the following procedure:

1. Stop the remote task.
2. Add the desired tables (as described in Adding Tables and/or Views to a Task).
3. Resume the remote task. The newly added tables will be marked as "Queued".
4. Reload the newly added tables in the local task (by selecting the tables and clicking the **Reload** icon in Monitor view).

For information on removing specific tables from a replication task, see Removing Specific Tables/Views from a Replication Task.

Note Adding tables to the remote task is not supported in Apply Changes (CDC-only) replication tasks. For more information on the available replication options, see Setting up Tasks.

Working with the File Channel Data Files

The File Channel stream data files are encoded in an internal binary format. For full-load operations, the File Channel binary files contain packed data records for each of the table records and an end-of-file (EOF) record. For change-processing operations, the file contains:

- A packed data record for each DDL and/or DML change.
- A **begin-load-table** record with the stream name that marks the beginning of table loading.
- A packed table-definition record with the table metadata. These records come before each DDL and **begin-load-table** record.

You do not need to work directly with the file-channel files, however if you find it necessary to work with them they are located in the File-Channel Directory Structure.

File-Channel Directory Structure

The file-channel directory contains the following files and folders:

- **s_msgs**: This folder contains messages sent from the source side to the replication server on the remote target side.

  Messages are removed from this folder at the source side when an acknowledgment message is received stating that the file was transferred successfully or possibly with a timeout.

  Messages are removed from this folder at the target side after they are read.

  This folder contains the following files:

  - **s_msgs/xxxxxxxx.fcm**: This file contains a JSON message from the source side to the target side.
  - **yyyymmddhhMMsss.mtd**: This file contains the captured tables list.
- **s_status**: This folder contains status updates from the source side to the target side. Status updates appear as a fixed name file that is periodically updated. This file lists the last processed target status file. It receives the t_status/cccccccc.fcs file. These files are deleted when the file-channel source endpoint finishes reading the file. You can configure the file-channel source to keep the files, if necessary. See Setting Advanced Connection Properties for more information.
- **t_status**: This folder contains status updates from the target side to the source side. Status updates appear as an infinite set of data files that are created according to a specific schedule. These files are sent from the target by the source. The folder contains also a fixed name file that is updated with the last created status file name. It contains the following file:

  - **t_status/cccccccc.fcs**: This is a file channel status file (.fcs) where the file name is a hexadecimal counter of length 8. These files will be transferred in order with the lower numbers transferred first. If you need to view them, you should order them by timestamp because alphabetical ordering will not be consistent with the hexidecimal name.

    File channel status files are deleted by the source after being read and by the target when source status file indicates that this file was already processed.

    You can configure the maximum amount of time that the files are kept before a new file is created as well as the maximum file size for each file. The minimum file size is 50 MB.

    For more information, see Setting Advanced Connection Properties.
- **streams/<stream-name>**: This folder contains stream sub-folder, one sub-folder per stream. A stream represents a finite or infinite set of data files being sent from the source to the target. The file channel allows creating and destroying named streams dynamically. For example, there can be a fixed-named stream `cdc` (streams/cdc) and there could be a dynamically created stream `loadXXXXXXXX` that can be removed at the source side when a status update from the target is received (for example, when processing completed) in the **t_status** folder.

You can configure the maximum number of streams and the maximum disc space for each stream. For more information, see Change Processing.

This folder contains the following file:

- **streams/<stream-name>/cccccccc.fcd**: This is a file channel data file (.fcd) where the file name is a hexadecimal counter of length 8. These files are processed at the target in order or in parallel depending on the case. However, the files are transferred in order with the lower numbers transferred first.

  File channel data files are deleted by the source when transferred successfully and by the target when processed.

  You can configure the maximum amount of time that the files are kept before being creating a new file and the maximum file size for each file. The minimum file size is 10 MB and the minimum time that a file is kept is 5 seconds.

Attunity Replicate Installation Requirements for the File Channel

To work with the file-channel endpoint, you must install Attunity Replicate anywhere on the network for each LAN that you are working with.

Security

When using the File Transfer Service, file-channel files are always transferred over an encrypted session.

The session is encrypted as follows:

The client and server create an AES-256 session key using the Diffie-Hellman key exchange protocol (using the OpenSSL library). After the key is created, all file transfers between the client and the server will take place over a secure and encrypted communication channel.

However, even though the session is encrypted, communication between the client and the server may still be susceptible to man-in-the-middle attacks. A man-in-the-middle in possession of the session key would be able to intercept any data transferred between the client and the server.

To eliminate man-in-the-middle attacks, a "shared password" needs to be provided when configuring the local and remote file channel endpoints. Once the session is established, both the client and the server use the shared password to re-key the session key during the next packet exchange, thereby preventing the original session key from being used for man-in-the-middle attacks.

To sum up:

1. Strong encryption is used regardless of whether a password was provided.
2. Providing a password eliminates the risk of a man-in-the-middle attack.

For more information about the File Transfer Service, see File Transfer Service.

Limitations

The following limitations apply:

- The File Channel endpoint does not support Full LOB mode.
- You cannot use the Full Load resume function if you are using the File Channel endpoint. To resume a Full Load operation, you must delete the original data and then run the task again.
- You must delete the File Channel folder before restarting an Apply Changes task.
- After modifying an existing transformation in a remote File Channel task, both the local and the remote File Channel tasks need to be *restarted* (by selecting the **Reload Target** run option in both tasks).
- Control tables defined for the Local File Channel task but not for the Remote File Channel task will not be created on the remote task's target endpoint.

  For information on defining Control Tables, see Control Tables.

Using the File Channel as a Source

The File Channel source endpoint is an Attunity Replicate endpoint that consumes and applies the contents of a file channel directory structure that was produced by a corresponding File Channel target endpoint.

This section contains the following topic:

- Setting General Connection Properties

Setting General Connection Properties

This section describes how to configure general connection properties. For an explanation of how to configure advanced connection properties, see **Setting Advanced Connection Properties** below.

To add the File Channel source to Attunity Replicate:

1. In **Tasks** view, click **Manage Endpoint Connections** toolbar button to open the **Manage Endpoint Connections** window.
2. In the **Name** field, type a name for your endpoint. This can be any name that will help to identify the endpoint being used.
3. In the **Description** field, type a description that helps to identify the information being replicated to the file. This is optional.
4. Select **SOURCE** as the endpoint **Role**.
5. Select **File Channel** as the endpoint **Type**.
6. Type the full path to the **Storage Folder** where the File Channel files will be created.

The default path when *not* using the File Transfer Service is:

`C:\Program Files\Attunity\Replicate\data\tasks\<task_name>`

If you are using the File Transfer Service, the default path is:

`C:\Program Files\Attunity\Replicate\data\endpoints\<file-channel_db_name>\fc`

Note The Replicate File Transfer Service always transfers the local file channel task's files *to the default directory* on the remote system (`C:\Program Files\Attunity\Replicate\data\endpoints\<remote_file-channel_db_name>\fc`). Consequently, if you are using the File Transfer Service, ensure that the default directory always has enough space for the incoming files.

For more information on using the File Transfer Service, see File Transfer Service and Using Advanced Properties for a File-Channel Source.

This folder should be in a location that is accessible from anywhere in the WAN you are working with.

Note

- You can use the **Advanced** tab to define specific properties and create a custom connect string. In this case, you do not need to enter information in this tab. For more information on using the **Advanced** tab, see Using Advanced Properties for a File-Channel Source.
- To determine if you are connected to the endpoint you want to use or if the connection information you entered is correct, click **Test Connection**.

  If the connection is successful a message in green is displayed. If the connection fails, an error message is displayed at the bottom of the dialog box.

  To view the log entry if the connection fails, click **View Log**. The server log is displayed with the information for the connection failure. Note that this button is not available unless the test connection fails.

7. Click **OK** to finish the setup and save the changes.

Using Advanced Properties for a File-Channel Source

You can set the following properties in the **Advanced** tab:

- **Input files are received via file transfer service**: Select this check box to receive the source input files using the Replicate File Transfer Service.
  - **Password**: The password that will be used to establish a secure connection with the File Channel Target.

    Important: When using the File Transfer Service, an agreed upon password is required in order to establish a secure connection between the File Channel Source

and the File Channel Target. Accordingly, the password specified in the File Channel Source settings and the password specified in the File Channel Target settings *must be identical*.

For more information about the File Transfer Service, see File Transfer Service.

- **Delete processed files**: Select this check box to delete the File Channel files after the data has been replicated to the target endpoint.

  You should clear this check box if other tasks need to use the files.

Internal Parameters

Internal parameters are parameters that are not exposed in the UI. You should only use them if instructed by Attunity Support.

To add internal Attunity Replicate parameters:

1. Click the **Internal Parameters** link.

   The **Internal Parameters** dialog box opens.
2. In the edit box, type the name of the parameter you need to add and then click it.
3. The parameter is added to the table below the search box with its default value.
4. Change the default value as required.
5. To reset the parameter value to its default, click the "Restore default value" icon at the end of the row.

Settings Summary

You can view a summary of your settings by clicking the **Setting Summary** link. This is useful if you need to send a summary of your settings to Attunity Support.

Using the File Channel as a Target

The File-Channel target endpoint is an Attunity Replicate endpoint that creates and maintains a file-based directory structure containing replication artifacts (task definitions, metadata, full load data, CDC data and status updates). This file channel directory structure is consumed by a corresponding File-Channel source endpoint in a different task and possibly in a remote location.

This section contains the following topic:

- Setting General Connection Properties

Setting General Connection Properties

This section describes how to configure general connection properties. For an explanation of how to configure advanced connection properties, see **Setting Advanced Connection Properties** below.

Note

- The **Type** is different depending on the type of file you are creating, however the information you enter is the same for all file types.
- All files are used as targets, however you can use an Attunity Replicate file as a source only after you created the file by loading data into it as a target.

To add the File Channel target to Attunity Replicate:

1. In the Attunity Replicate Console, click the **Manage Endpoint Connections** toolbar button to open the **Manage Endpoints Connections** dialog box. Then click the **New Endpoint Connection** button.
2. In the **Name** field, type a name for your endpoint. This can be any name that will help to identify the endpoint being used.
3. In the **Description** field, type a description that helps to identify the information being replicated to the file. This is optional.
4. Select **TARGET** as the endpoint **role**.
5. Select **File Channel** as the endpoint **Type**.
6. If you changed the default data folder during installation, type the full path to the **Storage Folder** (e.g. `D:\data\tasks\`) where the file is being created. Otherwise, you can leave this field empty. Note that this field will be ignored when the **Transfer files to remote file channel** option is enabled in the **Advanced** tab.

   Note

   - You can use the **Advanced** tab to define specific properties and create a custom connect string. In this case, you do not need to enter information in this tab. For more information on using the **Advanced** tab, see Setting Advanced Connection Properties.
   - To determine if you are connected to the endpoint you want to use or if the connection information you entered is correct, click **Test Connection**.

     If the connection is successful a message in green is displayed. If the connection fails, an error message is displayed at the bottom of the dialog box.

     To view the log entry if the connection fails, click **View Log**. The server log is displayed with the information for the connection failure. Note that this button is not available unless the test connection fails.

7. Click **OK** to finish the setup and save the changes.

Setting Advanced Connection Properties

You can set the following properties in the **Advanced** tab:

- **Max file size (KB)**: Click the arrows to select, or type the maximum file size (in kilobytes) allowed for the files created in the target.

- **Limit storage size to (MB)**: To allocate a specific amount of disk space to the File Channel files, enable this option and then specify the amount of disk space to set aside (using the arrows or by typing). When the limit is reached, Attunity Replicate will stop writing the files to the designated storage.
- **Max batching time interval (seconds)**: Click the arrows to select, or type the maximum time (in seconds) for files to be batched before being written in a single operation.
- **Transfer files to remote file channel**: Select this check box to transfer files to the File Channel Source (on the remote Attunity Replicate Server) using the Attunity Replicate File Transfer Service. This can dramatically improve transfer speeds when the source endpoint and the target endpoint are located on different LANs. For more information about the Attunity Replicate File Transfer Service, see File Transfer Service.
  - **Remote file transfer service host:** The host name or IP address of the computer on which the Attunity Replicate File Transfer Service is running.
  - **Remote file transfer service port:** The port on the remote computer through which the files will be transferred (from the storage folder to the remote file channel).
  - **Remote file transfer service endpoint name:** The name of the File Channel Source endpoint on the remote machine.
  - **Additional remote file channels:** When sending to multiple File Channel Source endpoints, specify the target destinations using the following format:

    ```
    file_channel_db_name@host:port,file_channel_db_name@host:port
    ```
  - **Max transfer streams:** The maximum number of streams to use when transferring the files. Adjust the number of streams as required to optimize transfer speeds.
  - **Password**: The password that will be used to establish a secure connection with the File Channel Source.

Important: When using the File Transfer Service, an agreed upon password is required in order to establish a secure connection between the File Channel Source and the File Channel Target. Accordingly, the password specified in the File Channel Target settings and the password specified in the File Channel Source(s') settings *must beidentical*.

Internal Parameters

Internal parameters are parameters that are not exposed in the UI. You should only use them if instructed by Attunity Support.

To add internal Attunity Replicate parameters:

1. Click the **Internal Parameters** link.

   The **Internal Parameters** dialog box opens.
2. In the edit box, type the name of the parameter you need to add and then click it.
3. The parameter is added to the table below the search box with its default value.
4. Change the default value as required.

5. To reset the parameter value to its default, click the "Restore default value" icon at the end of the row.

Settings Summary

You can view a summary of your settings by clicking the **Setting Summary** link. This is useful if you need to send a summary of your settings to Attunity Support.

11 | Customizing Tasks

This section describes how to customize a replication task. For example, you can create new tables or columns for the target endpoint or select only some of the data from each column to be replicated. This is done using transformations and filters.

Note Although the descriptions in this section only refer to tables, the procedures described herein are applicable to views as well. When a transformation is defined for a view, the word "View(s)" appears in the UI instead of the word "Table(s)".

In this chapter:

For more information about replication tasks, see Replication Tasks.

Table Settings

In the **Table Settings** dialog box, you can define how the data for each individual table/view is replicated to the target.

To open the Table Settings dialog box:

1. Open the task you are working with. For information on opening a task, see Editing a Replication Task.
2. In Designer view, on the right, select the table on which you want to perform the transformation.
3. Click **Table Settings**. If the table you want to perform the transformation on was defined by creating a table selection pattern, select the **Full Table List** tab.

   For information on how to define table selection patterns, see Creating Table/View Selection Patterns.
4. In the **Table Settings** window, perform any of the following tasks:
   - Carrying out General Tasks for a Single Table/View
   - Defining Transformations for a Single Table/View
   - Using Filters
5. Click **OK** to close the **Table Settings** window.

6. Click **Save** to preserve the table and column information for this task.
7. To revert changes you made to tables to their default values, click **Reset Table Defaults** at the bottom left of the **Table Settings** window. This option is available in all tabs.

   That this changes the data for all columns in the table to their default and removes any calculated columns that were added.

   Note This option is only available for tables with changes. Modified tables include the word **(changed)** in the table list.

Carrying out General Tasks for a Single Table/View

Note Although the descriptions in this section only refer to tables, the procedures describe herein are applicable to views as well. When a task is being performed for a view, the word "View(s)" will appear in the UI instead of the word "Table(s)"

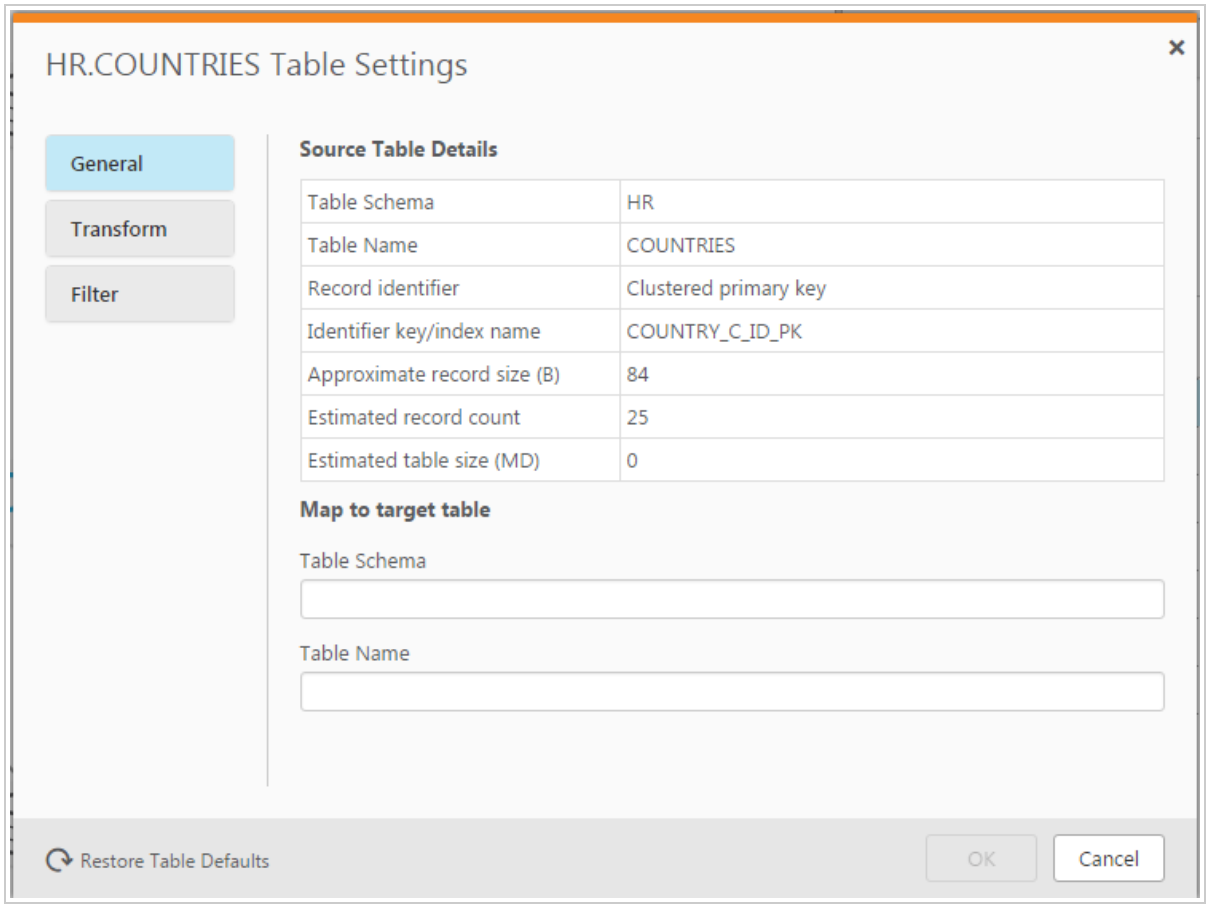
The **General** tab in the Table Settings window displays basic information about the selected table and allows you to define new names for the table/schema on the target. See figure Carrying out General Tasks for a Single Table/View to view an example of this information.

To edit the General table settings:

1. Open the Table Settings window.
2. Click **General** on the left side of the window.

   The following figure shows the information in the **General** tab of the **Table Settings**

window.



To rename the table or table schema:

In the **Map to target table** section, the following options are available:

- **Table Schema:** Specify the schema in which you want the table to be created on the target.
- **Table Name**: Specify a new name for the table on the target.

Defining Transformations for a Single Table/View

Note Although the descriptions in this section only refer to tables, the procedures describe herein are applicable to views as well. When a transformation is defined for a view, the word "View(s)" will appear in the UI instead of the word "Table(s)".

This section describes how to define data transformations. Data transformations are performed when the task is run. They are optional. If you do not define any transformations, the data is replicated "as is" from the source to the target.

Attunity Replicate lets you make the following changes to the tables and columns:

- Rename any column for the target table
- Delete a target column
- Change the data type and/or the length of any target column
- Add additional target columns
- Designate which target columns (i.e. segments) will comprise the Unique Index
- Recalculate the data

Limitations

Transformations are subject to the following limitations:

- They are not supported for calculating columns of Right-to-Left languages.
- They cannot be performed on columns that have a pound character (#) in their name.
- The only supported transformation for LOB/CLOB data types is to drop the column on the target.

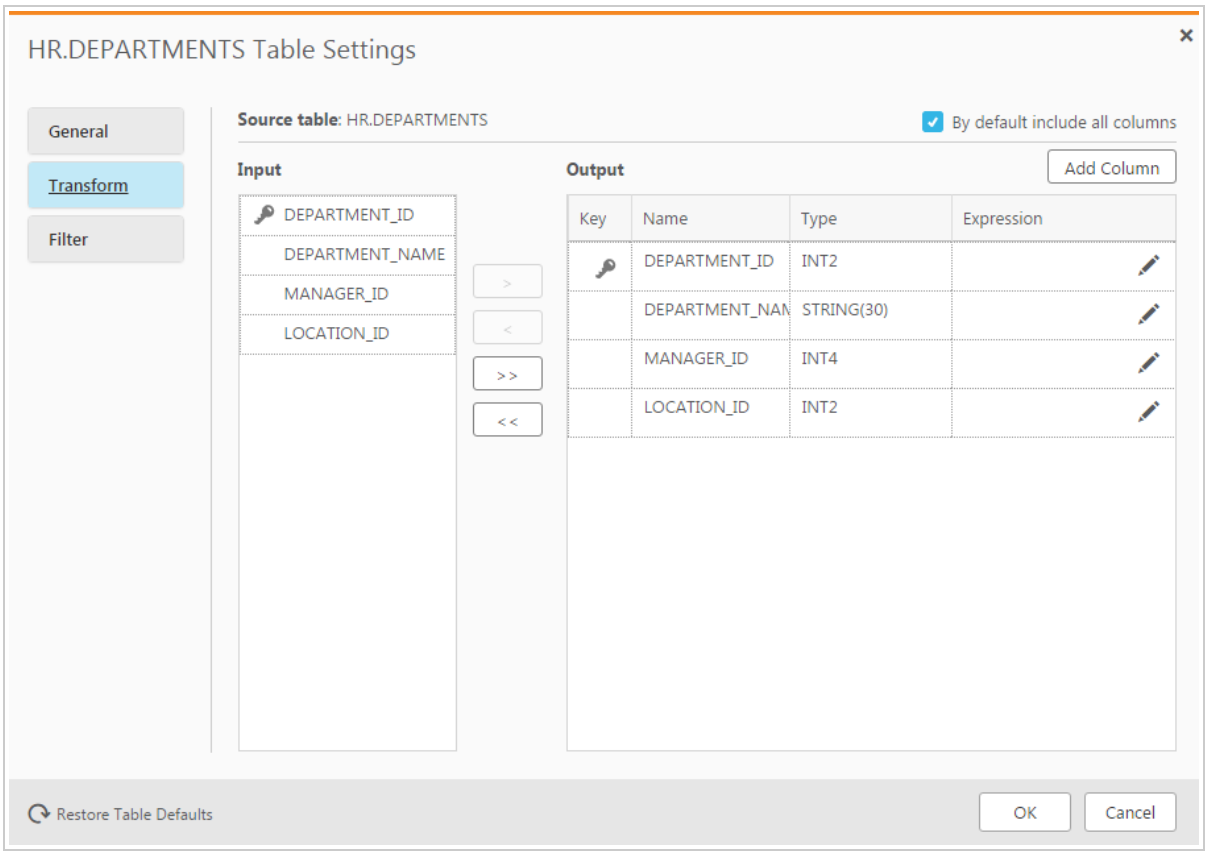
You can use the method described here for transformations that are specific to a single table or a few tables in your task. To make a similar change over multiple tables, see Defining Global Transformations.

For an explanation of how to configure transformations, see Using the Transform Tab.

To define a data transformation for a single table:

1. Select the table you want to transform and open the Table Settings window.
2. Click **Transform** on the left side of the window.

   The following figure shows the information in the **Transform** tab of the **Table Settings** window.

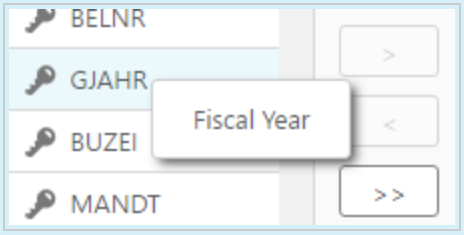


Using the Transform Tab

The **Transform** Tab in the **Table Settings** window has the following components:

- **Input**: This lists the columns on which you can perform transformations.

  Note When creating a transformation for the SAP Application source endpoint, you can hover your mouse cursor over an **Input** column to see a tooltip with the table's actual name:

- **Output**: This table shows the defined output for the columns in the table where you are performing the transformations. It contains the following columns:
  - **Key**: This indicates whether the column is a segment of the Unique Index. A key icon is displayed next to columns that are segments of the Unique Index. Click the column

to add and remove keys.

- **Name**: The name of the column. To change the name of the column, select the field with the column name you want to change and type a new name in this column if you want to change the name of the column or if the column is calculated (added to the table). See the table Transformation Actions for more information.
- **Type**: The data type for the column. To change the data type for the column, select the field with the data type you want to change and select a new data type. See the following table for more information.
- **Expression**: An expression using SQLite operators to define the data in the column. For information on how to create an expression, see the following table.

The following table describes the actions you can carry out in the Transform Table window.

Table 11.1 | Transformation Actions

To	Do This
Rename a column	Select the **Name** column for the table column you want to change. Type in a new name. The top right corner turns blue when the name is changed. To view the original name, hover the mouse pointer over the field and the original name is displayed.
Change the data type for a column	Select the **Type** column for the table column you want to change and select a new data type from the drop-down list. Make sure that the data type you select is compatible with the data in that column. For a description of Attunity Replicate data types, see Replicate Data Types. For information about data-type mapping from the native endpoint to Attunity Replicate data types, see the chapter for the endpoint you are using. For a list of supported databases, see Supported Platforms and Endpoints .
Add a new column	Click **Add Column** to add a new column. When you add a column, the **Name** is blank and the **Type** is listed as `string(50)`. Type a name for the new column in the **Name** column. Click in the **Type** column and select a data type from the list.

Table 11.1 | Transformation Actions (Cont.)

To	Do This
Add an existing column	From the **Input** pane, select one or more columns and click the right facing arrow button. To add all of the columns, click the right-facing double arrow. **Note**: By default all tables columns are included in the **Output** list. To include only some of the columns clear the **By default include all columns** check box at the top of the **Transform** tab. This removes all of the columns from the list. You can then add back any existing column.
Delete a column	From the **Output** list, select the row with the column you want to delete and click the left-facing arrow button. To remove all columns, click the left-facing double arrow. Note that all the columns except for columns defined as a primary key are deleted.
Add/Remove a Unique Index segment to/-from a target column	A key icon indicates which target columns segments of the Unique Index. To add a Unique Index segment, click in the **Key** column to the left of target column to which you want to add the segment. A key icon will appear. To remove a Unique Index segment, click the key icon to the left of the target column from which you want to remove the segment. The key icon will disappear.
Recalculate the data for a column in the target endpoint	Click in the **Expression** column in the row with the table column you want to change the data for. Enter an expression using SQLite syntax. See Creating an Expression for Transformations and Using SQLite Syntax with Transformations for information on creating expressions. Once you add a calculated expression, you

Table 11.1 | Transformation Actions (Cont.)

To	Do This
	can test the expression. See Using the Expression Builder (for Filters, Transformations, and Global Transformations).
Change the Character Set for a specific input column **Note** Supported on IBM DB2 for iSeries and IBM DB2 for z/OS only.	This is required if a source character column is wrongly encoded. For example, if a source character column is described as encoded in `CCSID X`, but the data stored in that column is actually encoded in `CCSID Y`. In the **Input** table: 1. Click the relevant cell in the **Type** column and select **STRING** from the drop-down list. 2. Click the relevant cell in the **Character Set** column and then select the appropriate character set from the drop-down list. **Note** Modified cells will display a triangle in the top right corner. To see the original value, click the triangle.
Change the data type for a specific input column **Note** Supported on IBM DB2 for iSeries and IBM DB2 for z/OS only.	This is required if a source column is defined as character type but the data stored in that column is binary or vice versa. In the **Input** table, click the relevant cell in the **Type** column and then select either **STRING** or **BYTES** from the drop-down list as required. Note that if you select STRING, you can also change the character set as explained above. **Note** Modified cells will display a triangle in the top right corner. To see the original value, click the triangle.

For a description of the various list actions that you can perform, see List Actions.

Creating an Expression for Transformations

Use an expression to define the contents of a new or re-calculated column.

To create an expression:

1. In the **Transform** tab, select the row with the column for which you want to create an expression.

   or

   Click **Add Column** to add a new column.
2. Click the pencil icon in the **Expression** column.

   The **Expression Builder** opens.
3. Build an expression as described in Using the Expression Builder (for Filters, Transformations, and Global Transformations).

Using SQLite Syntax with Transformations

The following table lists the SQLite operators that are supported with transformations.

Table 11.2 | SQLite Operators used by Attunity Replicate

Operator	Description
||	Concatenate strings. `FIRST_NAME||LAST_NAME` `PHONE_NUMBER`||<Office Only> (adds the string Office Only to the telephone number).
+	Adds two values together. `DEPARTMENT_ID`+100 (adds 100 to each ID number). Any column used in an expression with this operator must be a numeric data type.
-	Subtracts a value from another value. `MANAGER_ID`-100 (subtracts 100 from each ID number). Any column used in an expression with this operator must be a numeric data type.
%	Uses the remainder of a division expression as the value. %`SALARY`/7 (Divides the value of the Salary column by 7 and uses any remainder from the expression as the column value).
/	Divides one value into another. `SALARY`/.16 (Divides the value of the Salary column by .16. **Note** If the two values in the division expression are integers (two NUMERIC columns with no digits after the

Table 11.2 | SQLite Operators used by Attunity Replicate (Cont.)

Operator	Description
	decimal) and the result is a fractional value, the result returned will be 0.
*	SALARY*.16 (Multiplies the value of the Salary column by .16. This could be used to calculate taxes that are subtracted from a salary).

For more information about SQLite syntax, see the SQLite documentation.

Using Filters

The filtering operation lets you create filters that define the information from a column to include in/exclude from a replication task. This lets you replicate only the specific data that you need.

In this section:

Filter Limitations

When creating a filter, the following limitations apply:

- Filters are not supported for calculating columns of Right-to-Left languages.
- Filters can only be applied to immutable columns.
- When a filter is created to exclude specific rows in a column, the specified rows will always be excluded, even if the rows that were initially excluded are later changed. For example, if you chose to exclude rows "1-10" in a column named "Age" and those rows were later changed to "11-20", the rows will continue to be excluded, even though the data is no longer the same.
- Filter cannot be applied to LOB columns.

Opening the Filter Tab

The Filter Table tab contains the following information:

- **Data Columns** list: This list contains a list of the columns for the table where you filtering data. You can use these to select the columns to use in the filtering operations.

  This list has the following tabs:

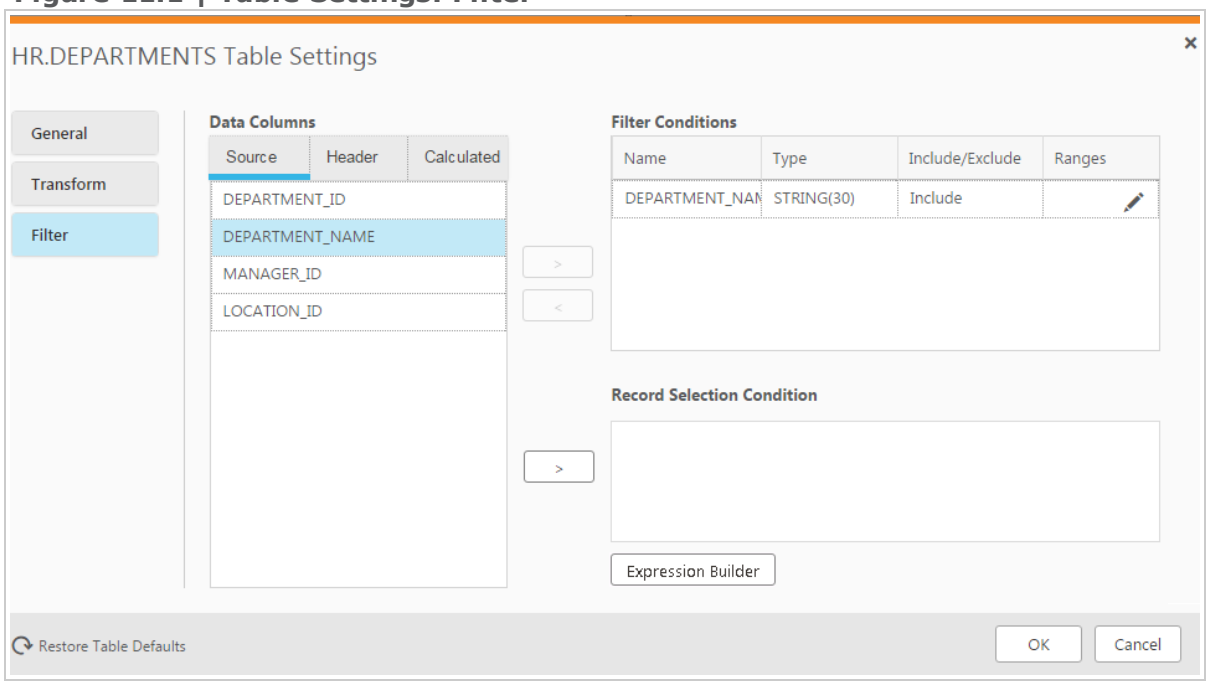
- **Source**: This tab lists the original source columns in the table.
- **Header**: This tab lists the available header columns. You can create filters using these columns and include them in expressions. For information on these header columns, see Header Columns.
- **Calculated**: This tab lists the columns added to the table. You add columns through transformations. For more information, see Defining Transformations for a Single Table/View.

- **Filter Conditions** table: This table has the following columns:
  - **Name**: The name of the column where you are filtering the data.
  - **Type**: The data type for the column.
  - **Include/Exclude**: Indicate whether to include or exclude the filtered data for this column.
  - **Ranges**: Click the button on the right of the **Ranges** field to open the Range Builder. For information on creating a value or ranges with the Range Builder, see Adding or Removing Filter Ranges.

    For more information on typing in the filter ranges manually, see Using SQLite Syntax with Filtering.
- **Record Selection Condition**: Enter a complex condition that can include multiple columns. The condition must evaluate to TRUE to be accepted. You can create a condition using SQLite operators or by Using the Expression Builder (for Filters, Transformations, and Global Transformations). For information on using the SQLite operators, see Creating a Record Selection Condition for One or More Columns.

The following figure is an example of the information in the **Filter** tab of the **Table Settings** window.

Figure 11.1 | Table Settings: Filter



To open the Filter tab:

1. Select the table you want to filter and then open the Table Settings window.
2. Click the **Filter** tab on the left side of the window.

Creating a Filter Condition for a Specified Column

You can create a simple condition for a single column in the table you are working with. You can include any combination of ranges or specific values in the filter and determine whether to include or exclude the defined data.

To create a filter condition:

1. Select a column from the data columns list and then click the right-facing arrow next to the **Filter Conditions** table.

   To remove the column, click on it in the **Filter Conditions** table and then click the left-facing arrow. Any data entered for this column in the **Include/Exclude** or Values columns is also deleted.
2. Click in the **Include/Exclude** column to select whether to include or exclude the data that meets this condition.
3. Click the Edit Ranges button in the **Ranges** column.
4. The **<Name> <Include|Exclude> Ranges** window opens. Continue from Adding or Removing Filter Ranges.

Creating a Record Selection Condition for One or More Columns

You can create a record selection condition manually and/or by using the Expression Editor.

When entering a string, you can use the following special characters:

- `%`: Matches any string of zero or more characters. For example, `Mc%` searches for every name that begins with **Mc** or `%bob%` includes every name that contains **bob**.
- `_`:Matches a single character (as a wildcard). For example: `'Sm_th'` includes names that begin with **Sm** and end with **th**, such as **Smith** or **Smyth**. To search for an underscore character, use [_]".
- `[..]`: Includes a range or set of characters. For example, `[CK]ars[eo]` includes names **Carsen**, **Karsen**, **Carson**, and **Karson** or `[M-Z]inger` includes all words that end in **inger** with the first letter between **M** and **Z**, such as Ringer, Singer, or Zinger.

For more information, see documentation on how to use Transact-SQL.

For information on what SQLite operators can be used to create Record Selection Condition filters, see Using SQLite Syntax with Filtering.

To create a record selection condition:

1. From the **Data Columns** list, select a source column, header column or calculated column and then click the arrow to the left of the **Record Selection Condition** pane.
2. Use SQLite operators, such as < or = to create the condition. Use any amount of strings or columns as you need to create a condition.

   For example `$EMPLOYEE_ID < 100 AND $SALARY > 100,000`

   In this case only rows that satisfy both of these conditions are replicated in the replication task.

   The following example provides an example using SQL search pattern strings. Only rows that satisfy this condition are replicated.

   ```
   $EMPLOYEE_NAME IS 'Sm_th'
   ```

To create a record selection condition using the Expression Builder:

- Click **Open Expression Builder**. This button is located directly under the record selection condition box. Follow the directions for creating an expression in the section Using the Expression Builder (for Filters, Transformations, and Global Transformations).

Adding or Removing Filter Ranges

You can add one or more values to the Ranges column using the Range Builder. Values that match *any* of the ranges in the list are included in the replication.

You can also delete a filter range using the Range Builder.

Note Filter ranges that you enter manually are also displayed in the Filter Builder. You can use the Filter Builder to delete them.

To use the Range Builder:

1. In the **Filter** tab of the Table Settings window, select a column to filter. For more information, see Using Filters.

2. Click the button to the right of the **Ranges** column.

   The **Ranges Builder** opens.

3. Click **Add Range**. Select any of the following from the drop-down list displayed.

   - **Equal to**: Select **Equal to** to enter a single value. The following is displayed in the range list.

     Equal to = [N]

     Click the **[N]** and type a value in the field that is displayed.

     When the value in the selected column equals the value you enter, the result is included or excluded in the replication task depending on the option selected in the **Include/Exclude** column.

   - **Between**: Click **Between** to enter a range of values. The following is displayed in the range list.

     Between [N] - [N]

     Click each **[N]** and type a value in the fields that are displayed.

     When the column contains the values between the two values entered, the result is included or excluded in the replication task depending on the option selected in the **Include/Exclude** column.

   - **Less than or equal to**: Select **Less than or equal to** and enter a maximum value. The following is displayed in the range list.

     Less than or Equal to =< [N]

     Click the **[N]** and type a value in the field that is displayed.

     When the value in the selected column is equal to or less than the value you enter, the result is included or excluded in the replication task depending on the option selected in the **Include/Exclude** column.

   - **Greater than or equal to**: Select **Greater than or equal to** and enter a minimum value. The following is displayed in the range list.

     Greater than or Equal to => [N]

     Click the **[N]** and type a value in the field that is displayed.

     When the value in the selected column is equal to or more than the value you enter, the result is included or excluded in the replication task depending on the option selected in the **Include/Exclude** column.

To delete a filter range from the Range Builder:

1. In the **Filter** tab of the Table Settings window, select the column with the filter condition you want to delete.
2. Click the button to the right of the **Ranges** column. The Ranges Builder opens.
3. Click the **X** next to the range you want to delete. The deleted range is removed from the list.

Using SQLite Syntax with Filtering

Attunity Replicate supports the following SQLite operators when creating Record Selection Condition filters.

Note You must put the ($) in front of each input as shown below.

Table 11.3 | SQLite Operators used by Attunity Replicate for Filtering

Operator	Description
<	Is less than. `$SALARY<100000`
<=	Is less than or equal to `$SALARY<=100000`
>	Is greater than `$SALARY>100000`
>=	Is more than or equal to `$SALARY>=100000`
=	Is equal to `$SALARY=100000`
!= or <>	Is not equal to `$SALARY!=100000`
IS	Is the same as `$HIRE_DATE IS 2014-09-29` IS functions the same as = unless one or both of the operands are NULL. In this case, if both operands are NULL, then the IS operator evaluates to 1 (true). If one operand is NULL and the other is not, then the IS operator evaluates to 0 (false).
IS NOT	Is not the same as `$HIRE_DATE IS NOT 2014-09-29` IS NOT functions the same as `!=` unless one or both of the operands are NULL. In this case, if both operands are NULL, the IS NOT operator evaluates to 0 (false). If one operand is NULL and the other is not, then the IS NOT operator evaluates to 1 (true).
AND	Both operands are true. `$MANAGER_ID AND EMPLOYEE ID >100`
OR	Either operand is true. `$MANAGER_ID OR EMPLOYEE ID >100`

For more information on how to use the SQLite syntax, see the SQLite documentation.

Defining Global Transformations

Use Global transformations to make similar changes to multiple tables, owners, and columns in the same task.

You may need to use this option when you want to change the names of all tables. You can change the names using wildcards and patterns. For example, you may want to change the names of the tables from `account_%` to `ac_%`. This is helpful when replicating data from an Microsoft SQL Server endpoint to an Oracle endpoint where the Microsoft SQL Server endpoint has a limit of 128 characters for a table name and the Oracle endpoint has a limit of 31 characters.

You may also need to change a specific data type in the source to a different data type in the target for many or all of the tables in the task. Global transformation will accomplish this without having to define a transformation for each table individually.

Note Table-specific transformations override global transformations. For example, you can define a global transformation that changes the data type for all tables from DATE to DATETIME(6) and then define another transformation for a specific table that changes the data type from DATE to STRING(50).

For information on defining a transformation for a specific table, see Defining Transformations for a Single Table/View.

This section includes the following topics:

Limitations for Global Transformations

The following limitations apply to global transformations:

- Transformations are not supported for columns with Right-to-Left languages.
- Transformations cannot be performed on columns that contain special characters (e.g. #, \, /) in their name.
- The only supported transformation for columns that are mapped to BLOB/CLOB data types (by Replicate) is to drop the column on the target.

Starting the New Transformation Rule Wizard

You define a rule for global transformation using the New Transformation Rule wizard. The transformation affects all of the tables in the task as you define them using the wizard.

To start the New transformation Rule wizard:

1. Open the task for which you want to create a global transformation.
   You can click **Open** above the Tasks list or double-click the task.
2. If you are not in the Designer mode, click **Designer** at the top right of the screen.
   For more information on the Task View and the Designer mode, see Designer Mode.
3. In Designer mode, click **Global Transformations**.
   The Global Transformation Rules window opens.
4. From the top of the **Global Transformation Rules** window, click **New Global Transformation**.
   The **New Transformation Rules** wizard opens.
5. Enter the information to define a global transformation rule. The first step is Selecting the Transformation Type.

Selecting the Transformation Type

In the **Which Global Transformation** step of the **New Transformation Rule** wizard, you define the type of transformation you want to be performed.

Note You can only create one rule for each transformation type *on the same object* (e.g. a column). If you create multiple rules for a single transformation type *on the same object*, only the last rule you create will be valid. For example, if you create the following rules (in order) to rename a schema:
Rename Schema: **Add Prefix**

Rename Schema: **Add Suffix**

-OR-

Rename Column: **Add Prefix**

Rename Column: **Add Suffix**

Only the second rule (adding a suffix) will be executed.

To select the transformation type:

1. Enter a name for the rule.
   The name cannot exceed 32 characters, contain non-Latin characters, or container any of the following characters: \/:*?"<>|
2. Select one of the following:
   - **Rename schema**: Select this if you want to change the schema name for multiple tables. For example, if you want all `HR` tables to be renamed `PERS`.
   - **Rename table**: Select this if you want to change the name of multiple tables. For example, if you want all tables named `SALARY` to be called `WAGES`.

- **Rename column**: Select this if you want to change the name of multiple columns. For example, if you want to change all columns with word `MINIMUM` to `MIN`.
- **Add column**: Select this if you want to add a column with a similar name to multiple tables.
- **Drop column**: Select this if you want to drop a column with a similar name from multiple tables.
- **Convert data type**: Select this if you want to change a specific data type to a different one across multiple tables. For example, if you want to change all Integer data types to a string.

3. Click **Next** to proceed to the Under what Conditions to Transform step.

Under what Conditions to Transform

In the **Under what conditions to transform?** step of the **New Transformation Rule** wizard, you define to which tables the transformation rule is applied. For example, you can apply the rule to all tables that contain the word `SALARY` as part of its name.

Note The options displayed in this screen depend on the Transformation Type selected.

The following table describes the available options.

Table 11.4 | Apply transformation rule if...

Option	Available when transformation type is:	Description
Schema name is like `%`	Always	Leave the `%` sign to include all schemas in your global transformation. Click the `%` sign to add a filter. In this case you can enter any name combination to include only that schema in your global transformation rule. For example, enter `HR` to include only tables that have the schema `HR`. You can use the `%` sign as a wildcard. For example, `H%` includes all tables with a schema that begins with the letter H, such as `HR`, `HELLO`, or `HQ`. The `%` wildcard can be used in any position. For example, if you use it at the beginning, `%H`, then all table names that end in H are included in the transformation rule. The `%` can also be used in a middle position.

Table 11.4 | Apply transformation rule if... (Cont.)

Option	Available when transformation type is:	Description
		Note If you are using an Oracle target, you must enter a schema that exists on the target endpoint. Attunity Replicate does not create new schemas on an Oracle endpoint. If you want to use a new schema for the target, create the schema on the Oracle endpoint before running the task. For more information, see Using Oracle as a Target.
Table name is like %	Always	Leave the % sign to include all table names in your global transformation rule. Click the % sign to add a filter. In this case you can enter any name combination to include only tables with that specific name in your global transformation rule. You can use the % sign as a wildcard. For example, J% includes all tables with a name that begins with the letter J, such as JOBS, JOBS_HISTORY, or JACKSONVILLE. The % wildcard can be used in any position. For example, if you use it at the beginning, %H, then all table names that end in H are included in the transformation rule. The % can also be used in a middle position.
Column name is like %	Rename Column Drop Column Convert Data Type	Leave the % sign to include all column names in your global transformation rule. Click the % sign to add a filter. In this case you can enter any name combination to include only columns with that specific name in your global transformation rule. You can use the % sign as a wildcard. For example, N% includes all columns with a name that begins with the letter N, such as NAME, NAME_FIRST, or NAME_LAST. The % wildcard can be used in any position. For example, if you use it at the beginning, %IES, then all column names that end in with the string "IES" are included in the transformation rule. The % can also be used in a middle position.
Data type is	Convert Data Type	Select a new data type from the drop-down list. Make sure that the data type you select is compatible with the data in that column.

Table 11.4 | Apply transformation rule if... (Cont.)

Option	Available when transformation type is:	Description
		For a description of Attunity Replicate data types, see Replicate Data Types. For information about data type mapping from the native endpoint to Attunity Replicate data types, see the chapter for the endpoint you are using. For a list of endpoints supported by Attunity Replicate, see Supported Platforms and Endpoints .

After you complete defining the transformation rule definitions, click **Next** to go to the Defining the Transformation Rule step.

Note If the global transformation type you are defining is **Drop Column**, you do not need to create a Transformation Rule. In this case, click **Finish** to add the rule to the Global Transformation Rules list.

Defining the Transformation Rule

In the **How to transform** step, you define what happens to the objects that the transformation rule is applied to. For example, you can define a new name for the affected objects or add a prefix to the table names. For more information on defining the affected tables, see Under what Conditions to Transform.

You define the rule to be carried out using the options on this page. Limitations for Transformation Rules apply. See the section for any of the following transformation types you are using:

- Rename Schema
- Rename Table
- Rename Column
- Add Column
- Drop Column
- Convert Data Type

When done, click **Next**.

Limitations for Transformation Rules

The following limitations apply to transformation rules:

- Transformations are not supported for columns with Right-to-Left languages.
- Transformations cannot be performed on columns that contain special characters (e.g. #, \, /) in their name.

- The only supported transformation for columns that are mapped to BLOB/CLOB data types (by Replicate) is to drop the column on the target.

Note The options displayed in this screen depend on the Transformation Type selected.

Rename Schema

If your transformation type is **Rename Schema**, you can do the following:

- Rename schema to (string)
- Add a Prefix or Suffix
- Remove a Prefix or Suffix
- Replace a Prefix or Suffix with Different Characters
- Convert schema name to uppercase
- Convert schema name to lowercase
- Rename schema (expression)

Rename schema to (string)

Use the **Rename schema to: [string]** option to change the name of all table schemas that you defined in the Under what Conditions to Transform step to a different name. For example, if you have a schema called `Human_Resources` and want to change all instances of this name to `HR` then enter the string `HR`. You can enter any string in this field.

Add a Prefix or Suffix

Use the **Add a prefix or suffix** option to add additional characters to the beginning or end of the schema name for all schemas that fit the definition you created in the Under what Conditions to Transform step. For example, if the schema name is HR, you can add a suffix, such as `TAR` or `_TAR` to the schema name for all tables with that schema name. In this case, the resulting schema name will be `HRTAR` or `HR_TAR`.

Note If you are using Oracle as your target endpoint, Attunity Replicate does not create a new schema. Therefore, the schema name that is the result of replacing a prefix or suffix with a different string of characters must exist in the Oracle target endpoint. If the resulting schema name does not exist, you must create the schema in the Oracle endpoint before carrying out this task.

For more information, see Limitations for using Oracle as a Source or Target.

To globally add a prefix or suffix

1. Select **Add <Prefix/Suffix> Insert Characters to matching schema names**.
2. Click the word **Prefix** or **Suffix** and select one of these two from the list.
3. Click **[string]** to activate the field.

4. Type the characters you want as the prefix or suffix. If you want to include an underscore or other legal character to separate the prefix/suffix from the original name, you must add it as part of the character string.
5. Click **Finish** to add the rule to the Global Transformation Rules list.

Remove a Prefix or Suffix

Use the **Remove a prefix or suffix** option to remove a string of characters from the beginning or end of a schema name for all schema that fit the definition you created in the Under what Conditions to Transform step.

For example, you can use this option to remove the letters `_REV` from the schema name for all tables in the schema `HR_REV`. In this case the schema name in the target will be `HR`.

Note If you are using Oracle as your target endpoint, Attunity Replicate does not create a new schema. Therefore, the schema name that is the result of replacing a prefix or suffix with a different string of characters must exist in the Oracle target endpoint. If the resulting schema name does not exist, you must create the schema in the Oracle endpoint before carrying out this task.

For more information, see Limitations for using Oracle as a Source or Target.

To globally remove a prefix or suffix

1. Select **Remove <Prefix/Suffix> Insert Characters from matching schema names**.
2. Click the word **Prefix** or **Suffix** and select one of these two from the list.
3. Click **[string]** to activate the field.
4. Type the characters you want to remove. If you want to remove an underscore or other legal character from the original name, you must add it as part of the character string.
5. Click **Finish** to add the rule to the Global Transformation Rules list.

Replace a Prefix or Suffix with Different Characters

Use the **Replace a prefix or suffix** option to replace a string of characters with a different string of characters. You determine whether to replace the characters at the beginning or end of a schema name for all schema that fit the definition you created in the Under what Conditions to Transform step.

For example, you can use this option to replace the letters `_ORIG` with `_REPL` in the schema name for all tables in the schema `HR_ORIG`. In this case the schema name in the target will be `HR_REPL`.

Note If you are using Oracle as your target endpoint, Attunity Replicate does not create a new schema. Therefore, the schema name that is the result of replacing a prefix or suffix with a different string of characters must exist in the Oracle target endpoint. If the resulting schema name does not exist, you must create the schema in the Oracle endpoint before carrying out this task.

For more information, see Limitations for using Oracle as a Source or Target.

To globally replace a prefix or suffix

1. Select **Replace <Prefix/Suffix> Insert Characters by Insert Characters for all matching schema names**.
2. Click the word **Prefix** or **Suffix** and select one of these two from the list.
3. Click the first **[string]** to activate the field.
4. Type the characters from the existing (source) schema that you want to replace. If you want to include an underscore or other legal character from the original name in the string that you want to replace, you must add it as part of the character string.
5. Click the second **[string]** to activate the field.
6. Type the characters you want to use in the target. These characters replace the original (source) characters in the target.
7. Click **Finish** to add the rule to the Global Transformation Rules list.

Convert schema name to uppercase

Use the convert to uppercase option to convert all of the letters in a schema name to upper case. For example:

Schema_cat, becomes SCHEMA_CAT

schema_cat, becomes SCHEMA_CAT

sChEMa_Cat, becomes SCHEMA_CAT

To globally change the schema name to all uppercase

1. Select **Convert schema name to uppercase**.
2. Click **Finish** to add the rule to the Global Transformation Rules list.

Convert schema name to lowercase

Use the convert to lowercase option to convert all of the letters in a schema name to lower case. For example:

Schema_cat, becomes schema_cat

SCHEMA_CAT, becomes schema_cat

sChEMa_Cat, becomes schema_cat

To globally change the schema name to all uppercase

1. Select **Convert schema name to lowercase**.
2. Click **Finish** to add the rule to the Global Transformation Rules list.

Rename schema (expression)

Use the **Rename schema to [expression]** option to change the name of all table schemas that you defined in the Under what Conditions to Transform step to a different name. For example, if you have a schema called `Human_Resources` and want to change all instances of this name to `HR`.

Note If you are using Oracle as your target endpoint, Attunity Replicate does not create a new schema. Therefore, the schema name that is the result of replacing a prefix or suffix with a different string of characters must exist in the Oracle target endpoint. If the resulting schema name does not exist, you must create the schema in the Oracle endpoint before carrying out this task.

For more information, see Limitations for using Oracle as a Source or Target.

To globally change a schema name

1. Select **Rename schema to [expression]**
2. Click the button to the right of the **Rename schema** option to open the Expression Editor. For information on how to use the Expression Editor, see Using the Expression Builder (for Filters, Transformations, and Global Transformations). Then go to step 4.

   or

   Click **[expression]** to activate the field and continue with step 3.
3. Type an SQLite expression or a string (in quotes) to rename the schema. For example:
   - "New_Schema"
   - 'PREF_'||$SCHEMA_NAME_VAR||'_SUFF'

   You can use the following variables in the SQLite expression:
   - $SCHEMA_NAME_VAR
   - $TABLE_NAME_VAR
   - $COLUMN_NAME_VAR
   - $COLUMN_DATATYPE_VAR
4. Click **Finish** to add the rule to the Global Transformation Rules list.

Rename Table

If your transformation type is **Rename Table**, you can do the following:

- Rename table to (string)
- Add a Prefix or Suffix
- Remove a Prefix or Suffix
- Replace a Prefix or Suffix with Different Characters

- Convert table name to uppercase
- Convert table name to lowercase
- Rename table (expression)

Rename table to (string)

Use the **Rename table to: [string]** option to change the name of all tables that you defined in the Under what Conditions to Transform step to a different name. For example, if you have a table called EMPLOYEE and want to change all instances of this name to EMP then enter the string EMP. You can enter any string in this field.

Add a Prefix or Suffix

Use the **Add a prefix or suffix** option to add additional characters to the beginning or end of the table name for all tables that fit the definition you created in the Under what Conditions to Transform step. For example, if the table name is EMPLOYEES, you can add a suffix, such as TAR or _TAR to the table name for all tables with that table name. In this case, the resulting table name will be EMPLOYEESTAR or EMPLOYEES_TAR.

To globally add a prefix or suffix:

1. Select **Add <Prefix/Suffix> Insert Characters to matching table names**.
2. Click the word **Prefix** or **Suffix** and select one of these two from the list.
3. Click **[string]** to activate the field.
4. Type the characters you want as the prefix or suffix. If you want to include an underscore or other legal character to separate the prefix/suffix from the original name, you must add it as part of the character string.
5. Click **Finish** to add the rule to the Global Transformation Rules list.

Remove a Prefix or Suffix

Use the **Remove a prefix or suffix** option to remove a string of characters from the beginning or end of a table name for all tables that fit the definition you created in the Under what Conditions to Transform step.

For example, you can use this option to remove the letters _REV from the table name for all tables with the name EMPLOYEES. In this case the table name in the target will be EMPLOYEES.

To globally remove a prefix or suffix:

1. Select **Remove <Prefix/Suffix> Insert Characters from matching table names**.
2. Click the word **Prefix** or **Suffix** and select one of these two from the list.
3. Click **[string]** to activate the field.
4. Type the characters you want to remove. If you want to remove an underscore or other

legal character from the original name, you must add it as part of the character string.

5. Click **Finish** to add the rule to the Global Transformation Rules list.

Replace a Prefix or Suffix with Different Characters

Use the **Replace a prefix or suffix** option to replace a string of characters with a different string of characters. You determine whether to replace the characters at the beginning or end of a table name for all tables that fit the definition you created in the Under what Conditions to Transform step.

For example, you can use this option to replace the letters `_ORIG` with `_REPL` in the table names for all tables called `EMPLOYEE_ORIG`. In this case the table name in the target will be `EMPLOYEE_REPL`.

To globally replace a prefix or suffix:

1. Select **Replace <Prefix/Suffix> Insert Characters by Insert Characters for all matching schema names**.
2. Click the word **Prefix** or **Suffix** and select one of these two from the list.
3. Click the first **[string]** to activate the field.
4. Type the characters from the existing (source) schema that you want to replace. If you want to include an underscore or other legal character from the original name in the string that you want to replace, you must add it as part of the character string.
5. Click the second **[sting]** to activate the field.
6. Type the characters you want to use in the target. These characters replace the original (source) characters in the target.
7. Click **Finish** to add the rule to the Global Transformation Rules list.

Convert table name to uppercase

Use the convert to uppercase option to convert a table name to all upper case. For example:

Table_cat, becomes TABLE_CAT

table_cat, becomes TABLE_CAT

taBLe_Cat, becomes TABLE_CAT

To globally change the table name to all uppercase:

1. Select **Convert table name to uppercase**.
2. Click **Finish** to add the rule to the Global Transformation Rules list.

Convert table name to lowercase

Use the convert to lowercase option to convert a table name to all lower case. For example:

Table_cat, becomes table_cat

TABLE_CAT, becomes table_cat

taBLe_Cat, becomes table_cat

To globally change the table name to all lowercase:

1. Select **Convert table name to lowercase**.
2. Click **Finish** to add the rule to the Global Transformation Rules list.

Rename table (expression)

Use the **Rename table to [expression]** option to change the name of all tables that fit the definition you created in the Under what Conditions to Transform step. For example, if you have a table called `EMPLOYEE` and want to change all instances of this name as defined in the previous step it to `EMP`.

To change the table name:

1. Select **Rename table to: [expression]**
2. Click the button to the right of the **Rename table** option to open the Expression Editor. For information on how to use the Expression Editor, see Using the Expression Builder (for Filters, Transformations, and Global Transformations). Then go to step 4.

   or

   Click **[expression]** to activate the field and continue with step 3.
3. Type an SQLite expression or a string (in quotes) to rename the table. For example:
   - "New_Table"
   - 'PREF_'||$TABLE_NAME_VAR||'_SUFF'
3. You can use the following variables in the SQLite expression:
   - $SCHEMA_NAME_VAR
   - $TABLE_NAME_VAR
   - $COLUMN_NAME_VAR
   - $COLUMN_DATATYPE_VAR

Rename Column

If your transformation type is **Rename Column**, you can do the following:

- Rename column to (string)
- Add a Prefix or Suffix
- Remove a Prefix or Suffix
- Replace a Prefix or Suffix with Different Characters
- Convert column name to uppercase
- Convert column name to lowercase
- Rename Column (expression)

Rename column to (string)

Use the **Rename column to: [string]** option to change the name of all columns that you defined in the Under what Conditions to Transform step to a different name. For example, if you have a table called `SALARY` and want to change all instances of this name to `EMP` then enter the string `SAL`. You can enter any string in this field.

Add a Prefix or Suffix

Use the **Add a prefix or suffix** option to add additional characters to the beginning or end of the column name for all columns that fit the definition you created in the Under what Conditions to Transform step. For example, if the column name is `SALARY`, you can add a suffix, such as `TAR` or `_TAR` to the table name for all tables with that table name. In this case, the resulting table name will be `SALARYTAR` or `SALARY_TAR`.

To globally add a prefix or suffix:

1. Select **Add <Prefix/Suffix> Insert Characters to matching column names**.
2. Click the word **Prefix** or **Suffix** and select one of these two from the list.
3. Click the **[string]** to activate the field.
4. Type the characters you want as the prefix or suffix. If you want to include an under-score or other legal character to separate the prefix/suffix from the original name, you must add it as part of the character string.
5. Click **Finish** to add the rule to the Global Transformation Rules list.

Remove a Prefix or Suffix

Use the **Remove a prefix or suffix** option to remove a string of characters from the beginning or end of a column name for all columns that fit the definition you created in the Under what Conditions to Transform step.

For example, you can use this option to remove the letters `_REV` from the column name for all columns with the name `SALARY`. In this case the column name in the target will be `SALARY`.

To globally remove a prefix or suffix:

1. Select **Remove <Prefix/Suffix> Insert Characters from matching column names**.
2. Click the word **Prefix** or **Suffix** and select one of these two from the list.
3. Click **[string]** to activate the field.
4. Type the characters you want to remove. If you want to remove an underscore or other legal character from the original name, you must add it as part of the character string.
5. Click **Finish** to add the rule to the Global Transformation Rules list.

Replace a Prefix or Suffix with Different Characters

Use the **Replace a prefix or suffix** option to replace a string of characters with a different string of characters. You determine whether to replace the characters at the beginning or end of a column name for all columns that fit the definition you created in the Under what Conditions to Transform step.

For example, you can use this option to replace the letters `_ORIG` with `_REPL` in the column names for all columns called `SALARY_ORIG`. In this case the column name in the target will be `SALARY_REPL`.

To globally replace a prefix or suffix:

1. Select **Replace <Prefix/Suffix> Insert Characters by Insert Characters for all matching schema names**.
2. Click the word **Prefix** or **Suffix** and select one of these two from the list.
3. Click the first **[string]** to activate the field.
4. Type the characters from the existing (source) column that you want to replace. If you want to include an underscore or other legal character from the original name in the string that you want to replace, you must add it as part of the character string.
5. Click the second **[string]** to activate the field.
6. Type the characters you want to use in the target. These characters replace the original (source) characters in the target.
7. Click **Finish** to add the rule to the Global Transformation Rules list.

Convert column name to uppercase

Use the convert to uppercase option to convert a column name to all upper case. For example:

```
Column_cat, becomes COLUMN_CAT
column_cat, becomes COLUMN_CAT
coLUMnM_Cat, becomes COLUMN_CAT
```

To globally change the table name to all uppercase

1. Select **Convert column name to uppercase**.
2. Click **Finish** to add the rule to the Global Transformation Rules list.

Convert column name to lowercase

Use the convert to lowercase option to convert a column name to all lower case. For example:

```
Column_cat, becomes column_cat
column_cat, becomes column_cat
coLUMnM_Cat, becomes column_cat
```

To globally change the column name to all lowercase:

1. Select **Convert column name to lowercase**.
2. Click **Finish** to add the rule to the Global Transformation Rules list.

Rename Column (expression)

Use the **Rename column to [expression]** option to change the name of all tables that fit the definition you created in the Under what Conditions to Transform step. For example, if you have a column called `SALARY` and want to change it to `SAL`.

To change the column name:

1. Select **Rename column to: [expression]**
2. Click the button to the right of the **Rename column** option to open the Expression Editor. For information on how to use the Expression Editor, see Using the Expression Builder (for Filters, Transformations, and Global Transformations). Then go to step 4.

   or

   Click **[expression]** to activate the field and continue with step 3.
3. Type an SQLite expression or a string (in quotes) to rename the column. For example:
   - `"New_Column"`
   - `'PREF_'||$COLUMN_NAME_VAR||'_SUFF'`

   You can use the following variables in the SQLite expression:
   - `$SCHEMA_NAME_VAR`
   - `$TABLE_NAME_VAR`
   - `$COLUMN_NAME_VAR`
   - `$COLUMN_DATATYPE_VAR`

Add Column

When you add a column to multiple tables, you must provide a name, define the data type for the column and define the data that the column contains. The column that you define here is added to all tables that fit the definition you created in step Under what Conditions to Transform.

The following describes the information you must enter in the transformation rule page for adding a column.

- **Column name**: Click the **[string]** to activate the field. Type the name for the column in the field. A column with this name is added to all tables that fit the definition you created in step Under what Conditions to Transform.
- **Column data type**: Click the drop-down for a list of data types and select a new data type from the drop-down list. Make sure that the data type you select is compatible with the data in that column.

  For a description of available data types, see Replicate Data Types. For information about data type mapping from the native endpoint to Attunity Replicate data types, see

the chapter for the endpoint you use. For a list of supported databases, see Supported Platforms and Endpoints .

- **Computation expression**: Click the button to the right of this field to open the Expression Editor or type an expression using SQLite operators to define the data in the column.

  For information on how to use the Expression Editor to create an expression, see Using the Expression Builder (for Filters, Transformations, and Global Transformations).

  For more information on creating expressions, see Creating an Expression for Transformations and Using SQLite Syntax with Transformations.

Drop Column

This option does not require a transformation rule. For this option you complete the Global transformation Rule after the Under what Conditions to Transform step.

Convert Data Type

When you convert the data type for a column, use this page to select the data type you want to convert to. The data type that you define in this step is applied to all columns and tables that fit the definition you created in the Under what Conditions to Transform step. Make sure that the data type you select is compatible with the data in columns you defined.

To select a converted data type:

- Select an Attunity Replicate data type from the drop-down list.

For a description of Attunity Replicate data types, see Replicate Data Types.

For information about data type mapping from the native endpoint to Attunity Replicate data types, see the chapter for the endpoint you are using. For a list of supported databases, see Supported Platforms and Endpoints .

Viewing all Global Transformation Rules

The **Global Transformation Rules** dialog box lists the name and description of all notification rules that are defined for the Attunity Replicate instance you are working with. This is where you go to edit or delete a transformation rule.

In this section:

- Edit a Global Transformation Rule
- Delete a Global transformation Rule

Edit a Global Transformation Rule

You can make changes to any transformation rule.

Note You cannot change the name of a transformation rule

To edit a global transformation rule:

1. In the **Global Transformation Rules** dialog box, select the transformation rule you want to edit.
2. Click **Open** (at the top of the list).

   The **Edit Existing Transformation Rule** wizard opens.
3. Make any changes you need in the wizard. For information on how to work with each of the pages in the New transformation Rule wizard, see Defining Global Transformations.

Delete a Global transformation Rule

You can delete a Global transformation rule.

To delete a global transformation rule:

1. In the **Global Transformation Rules** dialog box, select the transformation rule you want to edit.
2. Click **Delete** (above the list).
3. When prompted for confirmation, click **OK**.

   The transformation rule is removed from the list and deleted from the system.

Using the Expression Builder (for Filters, Transformations, and Global Transformations)

The Attunity Replicate Expression Builder provides an easy way to build an expression. It provides you with easy access to the required elements for your expression without having to type out any information manually. You access the Expression Builder through the dialog boxes where you define Filters, Defining Transformations for a Single Table/View, and Global Transformations when you do any of the following:

- Rename Schema
- Rename Table
- Rename Column

The following topics describe the Expression Builder:

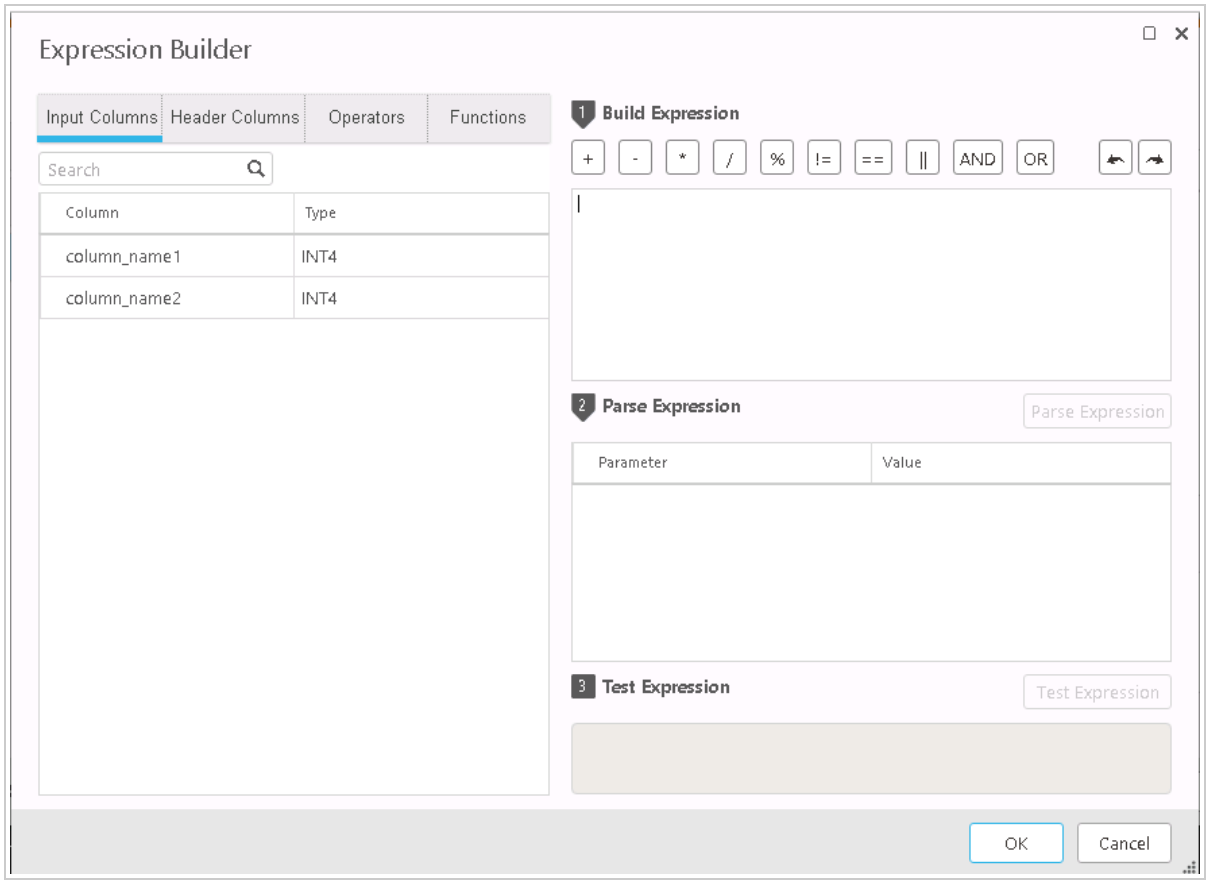
- Overview of the Expression Builder
- Build an Expression
- Evaluate an Expression
- Test an Expression
- Using Elements in the Expression Builder

Overview of the Expression Builder

The following is an example of the Expression Builder with its four main parts shown. The Expression Builder you are working with may look different depending on whether you

want to build an expression for a filter, a transformation, or a global transformation.

Figure 11.2 | Expression Builder for Filters, Transformations, and Global Transformations



The following sections describe what you can do in each part of the Expression Builder:

- **Elements Pane (on the left)**: This pane contains elements that you can add to an expression. Select elements and move them into the Expression Builder box to create the expression. For more information, see Build an Expression.

  The Elements Pane contains the following tabs:

  - **Metadata** (available only when working with Global transformations)
  - **Input Columns** (available only when working with transformations or filters)
  - **Header Columns** (for Global transformations, this tab is available only when you select **Add Column**)
  - **Operators**
  - **Functions**
- **Build Expression Panel**: The Build Expression Panel is where you put together the expression you are building. You move elements, such as columns or operators into the

box. You can also type all or part of an expression in this box. For more information, see Build an Expression.

- **Evaluate Expression Panel**: This panel displays the parameters for the expression. After you build the expression, click Evaluate to list the expression parameters. You can then enter a value or argument for each of the parameters. For more information, see Evaluate an Expression.

  The top part of the Expression panel contains the **Operator** toolbar. This toolbar contains the most common operators. Click the operator you want to use to add it to the expression. You can also add operators from the Element Pane, **Operators** tab.
- **Test Connection Expression Panel:** This panel displays the results of a test that you can run after you provide values to each of the parameters in your expression. For more information, see Test an Expression.

Build an Expression

The first step in using the expression builder is to build an expression. The expression that you build is displayed in the top section of the right pane. You can open the Expression when:

- You define Defining Transformations for a Single Table/View for a single table.
- You define Filters for a single table.
- You use the Global transformations dialog box to Rename Schema, Rename Table, Rename Column, or Add Column.

Note: To add operators to your expression, you can use the **Operator** tab in the Element pane or the Operator buttons at the top of the Build Expression panel or any combination of these. See Operators and Operator toolbar.

For example, to create an expression that will combine the first name and last name, do the following:

1. In the **Input Columns** tab add the FIRST_NAME column to the **Build Expression** box.
2. Click the concatenate (||) operator from the **Operator** bar at the top of the Build Expression box.
3. In the Input Columns tab add the LAST_NAME column into the **Build Expression** box.

To build an expression:

1. In the Elements Pane, select any element you want to include in your expression. For information on the elements you can use in an expression, see Functions.
2. Add an element to the **Build Expression** panel by selecting it and then clicking the arrow to the right of the element.
3. Continue to add elements as needed.

Operator toolbar

The Operator toolbar is above the Build Expression box. It contains the most common operators so you can easily add them to an expression.

The following operators are available in the Operator toolbar:

For information on these operators, see Operators.

To use the Operator toolbar:

1. Click the space in the Build Expression box where you want to add the operator.
2. Click the operator you want to add. It is added to the expression.

Evaluate an Expression

You can evaluate an expression to determine its parameters and to determine whether the expression is valid.

To evaluate an expression:

1. From the Expression Builder window, click Build an Expression.
2. Click **Evaluate**.

   If the expression is *not* valid, an error message is written in red at the bottom of the Expression Builder window.

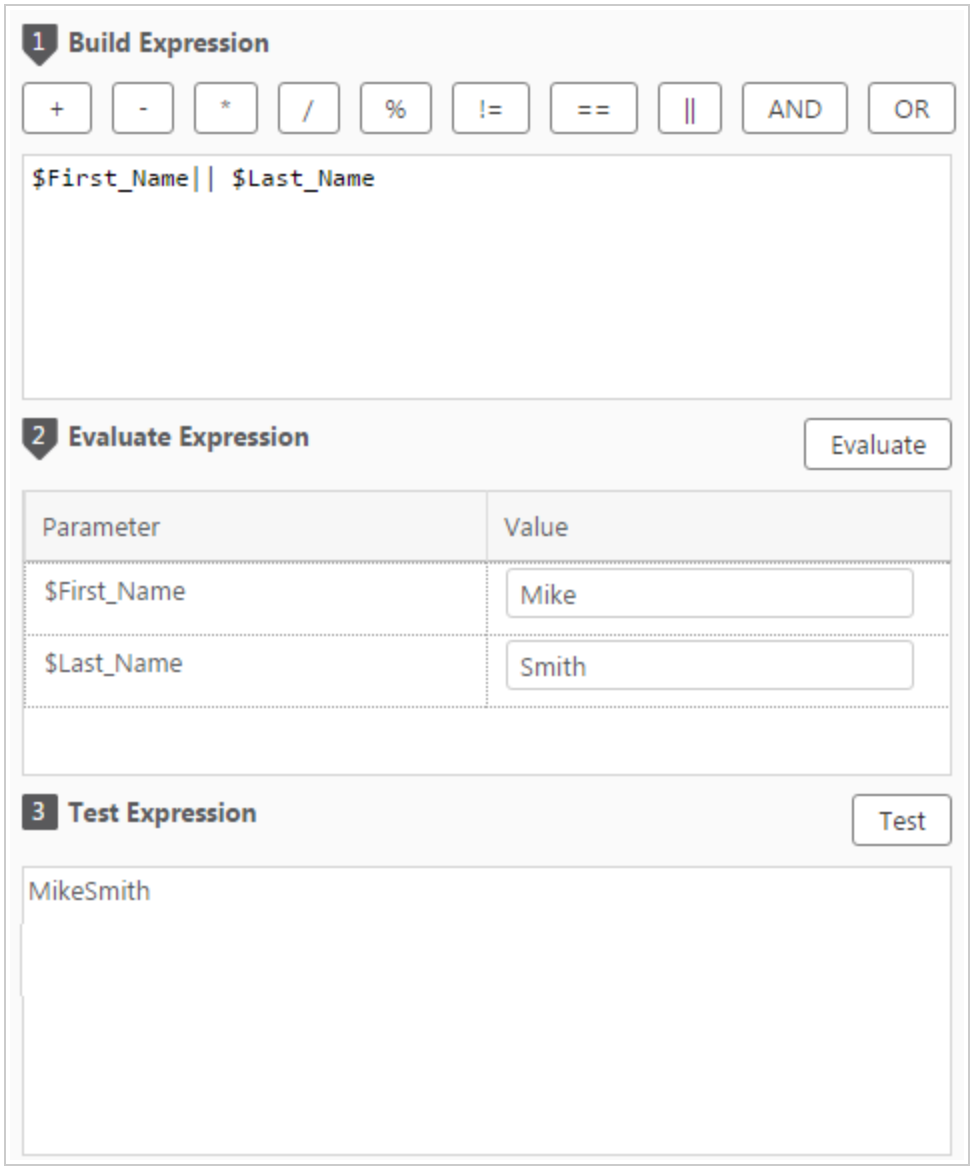
   If the expression is valid, the expression parameters are displayed in the **Parameter** column in the **Evaluate Expression** section. See the figure under Test an Expression.
3. Type a valid value for each of the parameters in the **Value** column to Test an Expression.

   For example, type `John` for the FIRST_NAME and `Smith` for the LAST_NAME in the **Value** column. Once you type in values, you can Test an Expression.

Test an Expression

You can use the Attunity Replicate Test procedure to display the results of a test expression. The following figure is an example of a built expression that is evaluated and contains a test result.

Figure 11.3 | Test Expression



To test an expression:

1. From the Expression Builder window, Build an Expression.
2. Click **Evaluate**. See Evaluate an Expression for more information.
3. View the parameters that are displayed. If your expression is not valid, an error message is displayed. See Evaluate an Expression.
4. Type values for each parameter then click **Test** to see the calculated expression.

   For example, type `John` for FIRST_NAME and `Smith` for LAST_NAME. The result displayed is `JohnSmith`. If you want a space between the words add it to the end of the FIRST_NAME value or the beginning of the LAST_NAME value.

Note: Testing calls to the `source_lookup` and `target_lookup` functions is not supported.

Using Elements in the Expression Builder

You can use the following types of elements to build expressions for transformations, filters, and global transformations. Select the appropriate tab to select the elements.

- Input Columns (transformations and Filters only)
- Metadata (Global Transformations Only)
- Operators
- Functions
- Header Columns

Input Columns (transformations and Filters only)

This section lists the columns for the table you are working with. The table you are working with is the table you selected when you opened the Table Settings dialog box.

Metadata (Global Transformations Only)

The **Metadata** tab contains the following variables that you can use in an expression:

- `AR_M_SOURCE_SCHEMA` - The name of the source schema.
- `AR_M_SOURCE_TABLE_NAME` - The name of the source table.
- `AR_M_SOURCE_COLUMN_NAME` - The name of a column in the source table.
- `AR_M_SOURCE_COLUMN_DATATYPE` - The data type of a column in the source table.

For example, to rename all columns named "metadata" to "*source_schema*.*table_name*", enter "metadata" in the **Column name is like** field (in the What to transform? screen) and then enter the following expression in the **Rename column to** field (in the How to transform? screen):

```
$AR_M_SOURCE_SCHEMA ||"."|| $AR_M_SOURCE_TABLE_NAME
```

Operators

The sections below describe the SQLite operators you can use to build an expression with the Expression builder. The Expression builder divides the operators into the following categories:

- Strings
- Logical
- Mathematical

Note All operator symbols must be preceded and followed by a space. For example, the expression for concatenating a first and last name should be specified like this:

```
FIRST_NAME || LAST_NAME
```

And not like this:

```
FIRST_NAME||LAST_NAME
```

Strings

You can use the following string:

```
||
```

Name: Concatenate strings.

Examples:

```
FIRST_NAME || LAST_NAME
```

```
PHONE_NUMBER || <Office Only>
```
(adds the string `Office Only` to the telephone number).

Logical

The following table describes the logical SQLite operators used by the Attunity Replicate Expression Builder.

Table 11.5 | Logical SQLite Operators used by Attunity Replicate Expression Builder

Operator	Description
!= or <>	Is not equal to `$SALARY!=100000`
IS	Is the same as `$HIRE_DATE IS 2014-09-29` IS functions the same as = unless one or both of the operands are NULL. In this case, if both operands are NULL, then the IS operator evaluates to 1 (true). If one operand is NULL and the other is not, then the IS operator evaluates to 0 (false).
IS NOT	Is not the same as `$HIRE_DATE IS NOT 2014-09-29` IS NOT functions the same as `!=` unless one or both of the operands are NULL. In this case, if both operands are NULL, the IS NOT operator evaluates to 0 (false). If one operand is NULL and the other is not, then the IS NOT operator evaluates to 1 (true).
IN	The IN operator takes a single scalar operand on the left and a vector operand on the right formed by an explicit list of zero or more scalars or by a single subquery. When the right operand

Table 11.5 | Logical SQLite Operators used by Attunity Replicate Expression Builder (Cont.)

Operator	Description
	of an IN operator is a subquery, the subquery must have a single result column. When the right operand is an empty set, the result of IN is false regardless of the left operand and even if the left operand is NULL. **Note** SQLite allows the parenthesized list of scalar values on the right-hand side of an IN operator to be an empty list but most other SQL endpoint engines and the SQL92 standard require the list to contain at least one element.
LIKE	The LIKE operator does a pattern matching comparison. The operand to the right of the LIKE operator contains the pattern and the left operand contains the string to match against the pattern. A percent symbol ("%") in the LIKE pattern matches any sequence of zero or more characters in the string. An underscore ("_") in the LIKE pattern matches any single character in the string. Any other character matches itself or its lower-/upper case equivalent. (By default SQLite only understands upper/lower case for ASCII characters. The LIKE operator is case sensitive by default for unicode characters that are beyond the ASCII range. For example, the expression 'a' LIKE 'A' is TRUE but 'æ' LIKE 'Æ' is FALSE.) LIKE can be preceded by the NOT keyword.
CASE	Evaluates a list of conditions and returns one of multiple possible result expressions. **Example 1:** WHEN $NEWEST = 'Y' THEN '1' ELSE '0' END **Example 2:** case length($month) when 2 then $year\|\|$month when 1 then $year\|\|0\|\|$month end
GLOB	The GLOB operator acts in the same way as the LIKE operator but uses the UNIX file globbing syntax for its wildcards. GLOB is case sensitive. GLOB can be preceded by the NOT keyword to invert the sense of the test. The infix GLOB operator is implemented by calling

Table 11.5 | Logical SQLite Operators used by Attunity Replicate Expression Builder (Cont.)

Operator	Description
	the function glob(Y,X) and can be modified by overriding that function.
MATCH	The MATCH operator is a special syntax for the match() application-defined function. The default match() function implementation raises an exception and is not really useful for anything. But extensions can override the match() function with more helpful logic.
REGEXP	The REGEXP operator is a special syntax for the regexp() user function. No regexp() user function is defined by default and so use of the REGEXP operator will normally result in an error message.
AND	Both operands are true. `$MANAGER_ID AND EMPLOYEE ID >100`
OR	Either operand is true. `$MANAGER_ID OR EMPLOYEE ID >100`
<<	Bitwise shift left. `x << n` A bitwise shift to the left of x by n bits.
>>	Bitwise shift right. `x >> n` A bitwise shift to the right of x by n bits.
&	Unary `and`
|	Unary `or`
<	Is less than. `$SALARY<100000`
<=	Is less than or equal to `$SALARY<=100000`
>	Is greater than `$SALARY>100000`
>=	Is more than or equal to `$SALARY>=100000`
= or ==	Is equal to `$SALARY=100000`

Mathematical

The following table describes the mathematical SQLite operators used by the Attunity Replicate Expression Builder.

Table 11.6 | SQLite Mathematical Operators used by the Attunity Replicate Expression Builder

Operator	Description
+	Adds two values together. `DEPARTMENT_ID`+100 (adds 100 to each ID number). Any column used in an expression with this operator must be a numeric data type.
-	Subtracts a value from another value. `MANAGER_ID`-100 (subtracts 100 from each ID number). Any column used in an expression with this operator must be a numeric data type.
%	Uses the remainder of a division expression as the value. %`SALARY`/7 (Divides the value of the Salary column by 7 and uses any remainder from the expression as the column value).
/	Divides one value into another. `SALARY`/.16 (Divides the value of the Salary column by .16. **Note**: If the two values in the division expression are integers (two NUMERIC columns with no digits after the decimal) and the result is a fractional value, the result returned will be 0.
*	`SALARY`*.16 (Multiplies the value of the Salary column by .16. This could be used to calculate taxes that are subtracted from a salary).

Functions

The sections below describe the SQLite functions you can use to build an expression with the Expression builder. The Expression builder divides the functions into the following categories:

- Strings
- LOBs
- Numeric
- NULL check
- Date and Time
- Data Enrichment
- Operation

Strings

The following table describes the string functions used by the Expression Builder in Attunity Replicate .

Table 11.7 | SQLite String Functions used by the Expression Builder

Function	Description
lower(x)	The lower(x) function returns a copy of string x with all characters converted to lower case. The default built-in lower() function works for ASCII characters only.
ltrim (x,y)	The ltrim(x,y) function returns a string formed by removing all characters that appear in y from the left side of x. If there is no value for y, ltrim(x) removes spaces from the left side of x.
replace (x,y,z)	The replace(x,y,z) function returns a string formed by substituting string z for every occurrence of string y in string x.
rtrim (x,y)	The rtrim(x,y) function returns a string formed by removing all characters that appear in y from the right side of x. If there is no value for y, rtrim(x) removes spaces from the right side of x.
substr (x,y,z)	The substr(x,y,z) function returns a substring of input string x that begins with the y-th character and which is z characters long. If z is omitted then substr (x,y) returns all characters through the end of the string x beginning with the y-th. The left-most character of x is number 1. If y is negative then the first character of the substring is found by counting from the right rather than the left. If z is negative then the abs(z) characters preceding the y-th character are returned. If x is a string then characters indices refer to actual UTF-8 characters. If x is a BLOB then the indices refer to bytes.
trim(x,y)	The trim(x,y) function returns a string formed by removing all characters that appear in y from both sides of x. If there is no value for y, trim(x) removes spaces from both sides of x.
upper(x)	The upper(x) function returns a copy of string x with all characters converted to upper case.

LOBs

The following table describes the LOB functions used by the Expression Builder in Attunity Replicate .

Table 11.8 | SQLite Lob Functions used by the Expression Builder

Function	Description
hex(x)	The hex() function receives an argument as a BLOB and returns an upper-case hexadecimal string version of the BLOB content.
randomblob (N)	The randomblob(N) function returns an N-byte BLOB that contains pseudo-random bytes. If N is less than 1 then a 1-byte random BLOB is returned.
zeroblob(N)	The zeroblob(N) function returns a BLOB that consists of N bytes of 0x00.

Numeric

The following table describes the numeric functions used by the Expression Builder in Attunity Replicate .

Table 11.9 | SQLite Numeric Functions used by the Expression Builder

Function	Description
abs(x)	The abs(x) function returns the absolute value of the numeric argument X. Abs (x) returns NULL if x is NULL. Abs(x) returns 0.0 if x is a string or BLOB that cannot be converted to a numeric value.
random()	The random() function returns a pseudo-random integer between -9223372036854775808 and +9223372036854775807.
round (x,y)	The round(x,y) function returns a floating-point value x rounded to y digits to the right of the decimal point. If there is no value for y, it is assumed to be 0.
max (x,y...)	The multi-argument max() function returns the argument with the maximum value, or returns NULL if any argument is NULL. The multi-argument max() function searches its arguments from left to right for an argument that defines a collating function and uses that collating function for all string comparisons. If none of the arguments to max() define a collating function, then the BINARY collating function is used. Note that max() is a simple function when it has two or more arguments but operates as an aggregate function if it has a single argument.
min (x,y...)	The multi-argument min() function returns the argument with the minimum value. The multi-argument min() function searches its arguments from left to right for an argument that defines a collating function and uses that collating function for all string comparisons. If none of the arguments to min() define a collating function, then the BINARY collating function is used. Note that min() is a simple function when it has two or more arguments but operates as an aggregate function if it has a single argument

NULL check

The following table describes the NULL check functions used by the Expression Builder in Attunity Replicate .

Table 11.10 | SQLite NULL Check Functions used by the Attunity Replicate Expression Builder

Function	Description
coalesce (x,y...)	The coalesce() function returns a copy of its first non-NULL argument, it returns NULL if all arguments are NULL. Coalesce() have at least two arguments.
ifnull (x,y)	The ifnull() function returns a copy of its first non-NULL argument, it returns NULL if both arguments are NULL. Ifnull() must have exactly two arguments. The ifnull() function is the same as coalesce() with two arguments.
nullif (x,y)	The nullif(x,y) function returns a copy of its first argument if the arguments are different and returns NULL if the arguments are the same. The nullif(x,y) function searches its arguments from left to right for an argument that defines a collating function and uses that collating function for all string comparisons. If neither argument to nullif() defines a collating function then the BINARY is used.

Date and Time

The following table describes the Date and Time functions used by the Expression Builder in Attunity Replicate .

Table 11.11 | SQLite Date and Time Functions used by the Attunity Replicate Expression Builder

Function	Description
`date`(timestring, modifier, modifier...)	Returns the date in the format YYYY-MM-DD.
`time`(timestring, modifier, modifier...)	Returns the time in the format HH:MM:SS.
`datetime`(timestring, modifier, modifier...)	Returns the date and time in the format YYYY-MM-DD HH:MM:SS.
`julianday` (timestring, modifier, modifier...)	The julianday() function returns the number of days since noon in Greenwich on November 24, 4714 B.C.
`strftime`(format, timestring, modifier, modifier...)	The strftime() routine returns the date formatted according to the format string specified as the first argument. It supports the following variables:

Table 11.11 | SQLite Date and Time Functions used by the Attunity Replicate Expression Builder (Cont.)

Function	Description
	%d: day of month %H: hour 00-24 %f: ** fractional seconds SS.SSS %j: day of year 001-366 %J: ** Julian day number %m: month 01-12 %M: minute 00-59 %s: seconds since 1970-01-01 %S: seconds 00-59 %w: day of week 0-6 sunday==0 %W: week of year 00-53 %Y: year 0000-9999 %%: %

Time strings can be in the following formats:

- YYYY-MM-DD
- YYYY-MM-DD HH:MM
- YYYY-MM-DD HH:MM:SS
- YYYY-MM-DD HH:MM:SS.SSS
- YYYY-MM-DDTHH:MM (T is a literal character that separates the date and time)
- YYYY-MM-DDTHH:MM:SS (T is a literal character that separates the date and time)
- YYYY-MM-DDTHH:MM:SS.SSS (T is a literal character that separates the date and time)
- HH:MM
- HH:MM:SS
- HH:MM:SS.SSS
- now (Converted to current date and time using UTC)
- DDDD.DDDD (The Julian day number expressed as a floating point value).

Data Enrichment

Data Enrichment functions allow the selected source tables to be augmented with data from other records located in either the source or target endpoints. Practical applications of data enrichment functions include code lookup or master record lookup (e.g. social security number lookup to find a person's name).

You can enrich the target tables with supplemental data retrieved from the source or target endpoint by defining a transformation on the table. For more information about defining transformations on a single table, see Defining Transformations for a Single Table/View.

Limitations

Amazon Redshift is not supported.

Data Enrichment Functions

The table below describes the source and target lookup functions, which can be used both for table transformations and for global transformations. For a description of the parameters available for these functions, see Input Parameters.

Table 11.12 | SQLite Data Enrichment Functions used by Expression Builder

Function	Description
`source_lookup (TTL,'SCHM','TBL','EXP','COND', COND_PARAMS)`	Use to retrieve additional data from the source endpoint.
`target_lookup (TTL,'SCHM','TBL','EXP','COND', COND_PARAMS)`	Use to retrieve additional data from the target endpoint.

Input Parameters

The possible input parameters for the lookup functions are described in the table below. For a usage example, see Functions.

Table 11.13 | Lookup Input Parameters for Data Enrichment Functions

Parameter	Description
`TTL`	`TTL` (Time to Live) is the amount of time the `'COND'` return value will be cached. Caching the `'COND'` return value improves performance by reducing the frequency that Attunity Replicate needs to access the source/target endpoint. As there is no default, you must specify a `TTL` value, which can be one of the following: `<SECONDS>` - The time to cache the `'COND'` return value in seconds. Specify a short caching time (e.g. `3`) for data that is frequently updated or a long caching time for data that rarely changes. `'NO_CACHING'`- Specify `'NO_CACHING'` if you do not want to cache the `'COND'` return value. This is recommended for data that is constantly updated (e.g. share prices). `'NO_EXPIRATION'`- For data that is never updated (e.g. a street name), specify `'NO_EXPIRATION'` to store the Functions return value permanently in the cache.

Table 11.13 | Lookup Input Parameters for Data Enrichment Functions (Cont.)

Parameter	Description
'SCHM'	The schema name.
'TBL'	The table on which to perform the lookup.
'EXP'	The expression to retrieve data from the lookup table. **Note:** The expression syntax must be native to the endpoint it accesses. The result should be a single column. Possible expressions include: `col1`, `col1+5`, `max(col1)`. **Note:** Full LOB columns are not supported. For information on including Limited-size LOB columns in the replication, see the description of the Metadata tab. Input Columns (transformations and Filters only), Header Columns, and Metadata (Global Transformations Only) can also be used in the expression and are evaluated before the lookup statement is performed against the endpoint.
'COND'	The condition for the lookup statement. **Note:** The condition syntax must be native to the endpoint it accesses. The COND is a single field referencing all required fields. **Example if the lookup table is located in Oracle:** 'Fieldname1=:1 and Fieldname2=:2 and Fieldname3 =:3' **Example if the lookup table is located in Microsoft SQL Server:** 'Fieldname1=? and Fieldname2=? and Fieldname3=?' Input Columns (transformations and Filters only), Header Columns, and Metadata (Global Transformations Only) can also be used in the expression and are evaluated before the lookup statement is performed against the endpoint.
COND_ PARAMS	Any parameters required by the COND parameter. The COND_PARAMS (condition parameters) is not a single field, but a list of fields. **Syntax:** $FIELDNAME1 , $FIELDNAME2 , $FIELDNAME3 **Full example:** source_lookup(10000 , 'HR' , 'DEPARTMENTS' , 'DEPARTMENT_NAME' ,

Table 11.13 | Lookup Input Parameters for Data Enrichment Functions (Cont.)

Parameter	Description
	'COMPANY_ID=? and DIVISION_ID=? and DEPT_ID=?' , **$COMP_ID , $DIV_ID , $DEPT_ID**)

Note To improve efficiency, the source/target lookup tables should be indexed for the specified lookup fields.

Data Enrichment Example

In the following example, Mike needs to add the `DEPARTMENT_NAME` column to the `HR.JOB_HISTORY` table. The `DEPARTMENT_NAME` column is located in the `HR.DEPARTMENTS` table in the source endpoint.

This is how the `HR.JOB_HISTORY` table appears before the column is added:

EMPLOYEE_ID	START_DATE	END_DATE	JOB_ID	DEPARTMENT_ID
102	2001-01-13 00:00:00	2006-07-24 00:00...	IT_PROG	60
101	1997-09-21 00:00:00	2001-10-27 00:00...	AC_ACCOUNT	110
101	2001-10-28 00:00:00	2005-03-15 00:00...	AC_MGR	110
201	2004-02-17 00:00:00	2007-12-19 00:00...	MK_REP	20
114	2006-03-24 00:00:00	2007-12-31 00:00...	ST_CLERK	50
122	2007-01-01 00:00:00	2007-12-31 00:00...	ST_CLERK	50
200	1995-09-17 00:00:00	2001-06-17 00:00...	AD_ASST	90
176	0001-09-09 09:30:25	2010-09-09 09:30...	SA_REP	60
176	2007-01-01 00:00:00	2010-09-09 09:30...	SA_MAN	60
200	2002-07-01 00:00:00	2006-12-31 00:00...	AC_ACCOUNT	90

This is how the `HR.JOB_HISTORY` table appears after the Full Load completes:

EMPLOYEE_ID	START_DATE	END_DATE	JOB_ID	DEPARTMENT_ID	DEPARTMENT_NAME
102	2001-01-13 00:00:00	2006-07-24 00:00...	IT_PROG	60	Warehouse
101	1997-09-21 00:00:00	2001-10-27 00:00...	AC_ACCOUNT	110	ProductManagement
101	2001-10-28 00:00:00	2005-03-15 00:00...	AC_MGR	110	ProductManagement
201	2004-02-17 00:00:00	2007-12-19 00:00...	MK_REP	20	HR
114	2006-03-24 00:00:00	2007-12-31 00:00...	ST_CLERK	50	Bookkeeping
122	2007-01-01 00:00:00	2007-12-31 00:00...	ST_CLERK	50	Bookkeeping
200	1995-09-17 00:00:00	2001-06-17 00:00...	AD_ASST	90	Management
176	0001-09-09 09:30:25	2010-09-09 09:30...	SA_REP	60	Warehouse
176	2007-01-01 00:00:00	2010-09-09 09:30...	SA_MAN	60	Warehouse
200	2002-07-01 00:00:00	2006-12-31 00:00...	AC_ACCOUNT	90	Management

To add the `DEPARTMENT_NAME` column, Mike needs to:

1. Create a new task and select the `HR.JOB_HISTORY` table for replication.
2. Apply a "New Column" transformation to the `HR.JOB_HISTORY` table. For more information on defining transformations, see Defining Transformations for a Single Table/View.

3. Open the **Expression Builder** and choose **Data Enrichment** from the **Functions** tab. For more information on the Expression Builder, see Using the Expression Builder (for Filters, Transformations, and Global Transformations).
4. Select the `source_lookup` function and configure it as follows (using the native syntax of the source endpoint):

   If the lookup table is located in Oracle:

   ```
   source_lookup(10000,'HR','DEPARTMENTS','DEPARTMENT_NAME',
   'DEPARTMENT_ID=:1',$DEPARTMENT_ID)
   ```

   If the lookup table is located in Microsoft SQL Server:

   ```
   source_lookup
   (10000,'HR','DEPARTMENTS','[DEPARTMENT_NAME]',
   '[DEPARTMENT]=?',$DEPARTMENT_ID)
   ```

   Where:

   - `10000` is the TTL parameter.
   - `HR` is the schema name.
   - `DEPARTMENTS` is the table name.
   - `DEPARTMENT_NAME` is the expression.
   - `DEPARTMENT_ID=:1` (or `?` on Microsoft SQL Server) is the condition.
   - `$DEPARTMENT_ID` is the condition parameter.
5. Run the task.

Operation

The following table describes the Operation functions used by the Expression Builder in Attunity Replicate .

Table 11.14 | SQLite Operation Functions used by the Attunity Replicate Expression Builder

Function	Description
`operation_indicator (value_on_delete, value_on_update, value_on_insert)`	When the `operation_indicator` function is invoked on its own or as part of an expression, records deleted from the source endpoint will *not* be deleted from the target endpoint. Instead, the corresponding target record will be flagged (with a user-provided value) to indicate that it was deleted from the source. The `operation_indicator` function also requires you to provide values to indicate records that were inserted or updated in the source endpoint. **Note:** The operation_indicator function is not supported on tables that do not have a Primary Key. **Note:** It is recommended to add a dedicated column for the flag values, for example, `OPERATION`. For an explanation of how to add a column, see

Table 11.14 | SQLite Operation Functions used by the Attunity Replicate Expression Builder (Cont.)

Function	Description

	Using the Transform Tab.
	To specify the function values:
	Replace `value_on_delete`, `value_on_insert` and `value_on_update` with the values that you want to appear in the target endpoint.
	Values should be formatted according to the corresponding column type.
	Example when the column type is INT4:
	`operation_indicator('1', '0', '0')`
	Example when the column type is `STRING`:
	`operation_indicator('Deleted', 'Updated', 'Inserted')`

Other Functions

The following table describes additional functions used by the Expression Builder in Attunity Replicate .

Table 11.15 | SQLite Functions used by the Attunity Replicate Expression Builder

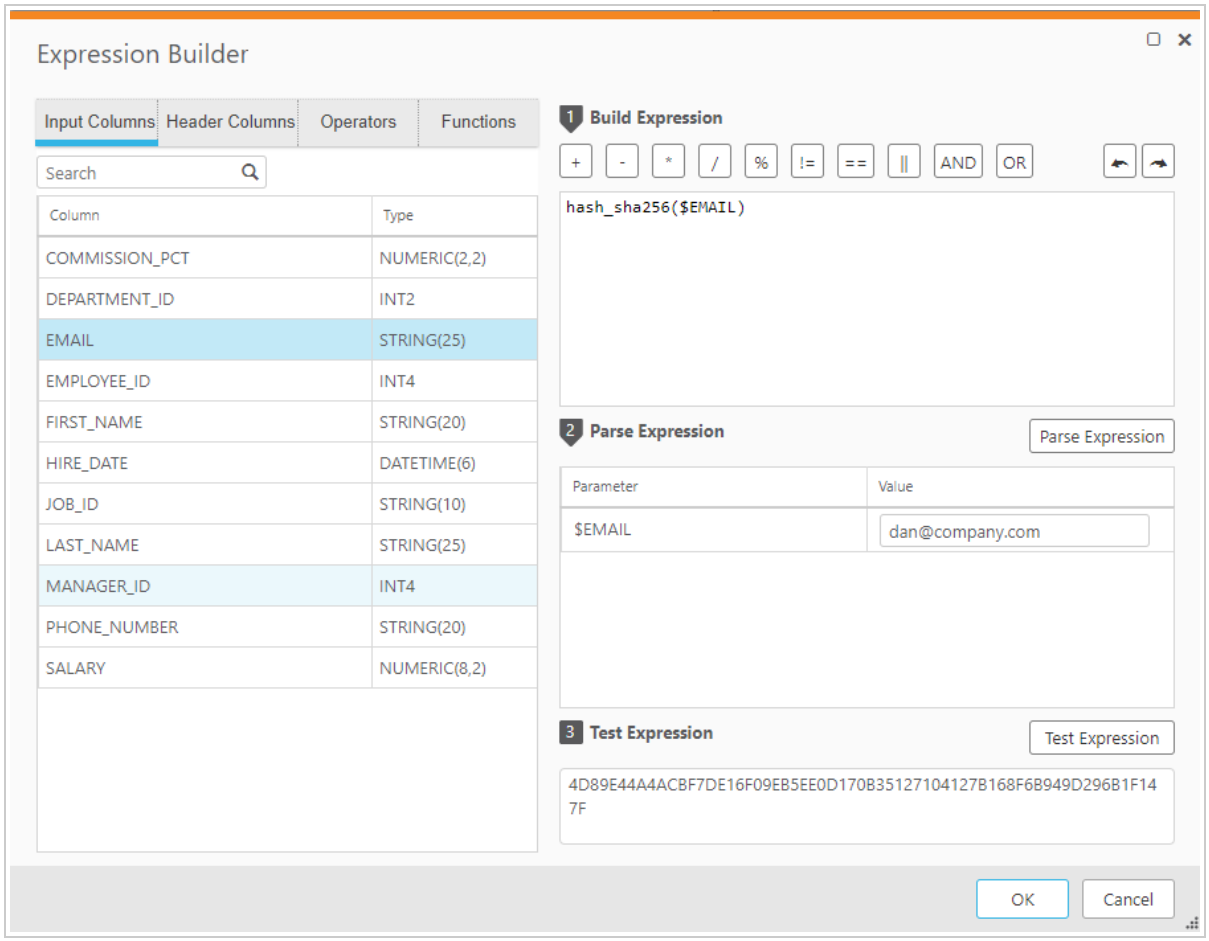
Function	Description
length(x)	For a string value x, the length(x) function returns the number of characters (not bytes) in x before to the first NULL character. If x is NULL then length(x) is NULL. If x is numeric then length(X) returns the length of a string representation of X.
like (x,y,z)	The like() function is used to implement the `"Y LIKE X [ESCAPE Z]"` expression. The ESCAPE (z) clause is optional. If there is a z clause, then the like() function is invoked with three arguments. Otherwise, it is invoked with two arguments.
typeof(x)	The typeof(x) function returns a string that indicates the datatype of the expression x: `null`, `integer`, `real`, `text`, or `BLOB`.

Hash

The Hash function generates a hash value for an inputted column (using the SHA-256 algorithm) and then returns the hex value of the generated hash value.

To use the function in an expression, add the `hash_sha256(x)` function to the **Build Expression** pane and then replace the "x" with the desired source column name (from the **Input Columns** tab).

The function is especially useful for masking sensitive information. In the expression below, for example, the Hash function has been used to obfuscate employees' email addresses.



Header Columns

By default, header columns for source tables are not replicated to the target. You can determine which, if any, header columns to replicate when you define a transformation by creating an expression that includes the header field.

You can create a filter using header field values. Header column filters are applied during change processing. See Using Filters for additional information.

Note The Header Column tab in the Expression builder is available for Filters and transformations. It is available for Global transformations only when you select **Add Columns**. See Selecting the Transformation Type.

The following table describes the header field columns.

Table 11.16 | Header Columns

Header Column Name	Value in Change Process	Value in Full Load	Data Type
AR_H_ STREAM_ POSITION	The stream position value from the source (For example, the SCN or LSN depending on the source endpoint).	Empty string	STRING
AR_H_ TIMESTAMP	Change timestamp	Current timestamp	DATETIME
AR_H_ COMMIT_ TIMESTAMP	Commit timestamp	Current timestamp	DATETIME
AR_H_ OPERATION	INSERT/UPDATE/DELETE	INSERT	STRING
AR_H_ USER	The user name, ID or any other information that the source provides about the change initiator. This header column is supported on the Microsoft SQL Server, IBM DB2 on iSeries (ARC), and Oracle (version 11.2.0.3 and higher) source endpoints only.	Empty	STRING

Task Settings

Each task has settings that you can configure according to your needs for replication. You configure the settings in the Task Settings dialog box.

To open the Task Settings dialog box:

1. Open the task you are working with if it is not displayed in the Attunity Replicate Console. For information on opening a task, see Editing a Replication Task.
2. Click **Task Settings**.
3. In the **Task Settings** dialog box, select the tab on the left with the task setting you want to configure. The following tabs are available:
   - Metadata
   - Bidirectional
   - Full Load Settings

- Change Processing
- Error Handling
- Logging

Metadata

When you click **Metadata** in the Task Settings dialog box, you can configure the **Target Metadata Settings** for a replication task.

Target Metadata

Target table schema: (if empty, use the schema from the source table): This will automatically add the owner prefix for the target endpoint to all tables if no source schema is defined.

> **Note** When replicating to a Hadoop target endpoint, the value specified in this field will be interpreted as a database name (as opposed to a schema name).

Replicate LOB columns (BLOB, CLOB and similar large object data types): Select this if the source tables include LOBs.

> **Note** LOB data types are supported only in tables that include a primary key.

If you select **Replicate LOB columns**, you must also select one of the following:

- **Allow unlimited LOB size**: If you select this option then enter a value for the following parameter:

  Chunk size (KB): Specify the size of the LOB chunks to use when replicating the data to the target.
- **Limit LOB size to**: If you select this option, specify the maximum permitted LOB size.

> **Important:** In some scenarios, tasks configured to replicate tables with multiple LOB columns may consume a large amount of memory. This is because Replicate allocates memory by multiplying the **Limit LOB size to** value by the Commit rate during full load value, the sum of which it multiplies by the number of LOB columns being replicated. So, for example, if LOB size is limited to 5 MB and the default commit rate is used (10000 events), a task replicating 6 LOB columns will consume 30 GB of memory.
>
> Should you encounter memory consumption issues and suspect that a combination of the above factors may be the cause, stop the task and lower the value in the Commit rate during full load field. Then resume the task. Repeat this process until acceptable performance/memory levels are reached.
>
> These instructions apply to Change Processing and Full Load tasks.

Note Changes to a column's LOB size while a task is running will not be reflected in the Change Table, unless the target tables are created by Attunity Replicate. In such cases, the task must be configured to drop and create the Change Table (the default) and the target tables need to be reloaded (after the LOB size has changed).

For more information on the Change Table, see Store Changes Settings. For information on reloading target tables, see Reload Target and Reload.

Control Tables

Control Tables provide information about the replication task as well as useful statistics that can be used to plan and manage both the current replication task and future replication tasks. Aside from the **Apply Exceptions** table which is always created, you can choose which Control Tables to create on the target.

Create control table in target using schema: Enter the endpoint schema for the target Control Tables. If you do not enter any information in this field, then the tables will be created in the default location in the endpoint.

Note When this field is left empty, the target endpoint is MySQL, and the Multiple Endpoints option is enabled, a default database named `attrep_control` will be created on the MySQL server. The selected control tables will be created in this database.

For more information on the Multiple Endpoints option, see Setting up a MySQL Database as a Target in Attunity Replicate.

Note When replicating to a Hadoop target endpoint, the value specified in this field will be interpreted as a database name (as opposed to a schema name).

Replication history time slot (minutes): The length of each time slot in the Replication History table. The default is 5 minutes.

Table Selection

In addition to the **Apply Exceptions** table (required), select which of the following Control Tables you want Attunity Replicate to create on the target endpoint: **Replication Status**, **Suspended Tables** and **Replication History**.

- **Replication Status:** Provides details about the current task including task status, amount of memory consumed by the task, number of changes not yet applied to the target and the position in the source endpoint from which Attunity Replicate is currently reading.
- **Suspended Tables:** Provides a list of suspended tables as well as the reason they were suspended.
- **Replication History:** Provides information about the replication history including the number and volume of records processed during a replication task, latency at the end of a CDC task, among others.

- **Change Data Partitions:** The **attrep_cdc_partitions** table contains records of partitions created on the target database when Change Data Partitioning is enabled for a Replicate task. You can use this information to identify partitioned data that needs to be further processed.
- **DDL History:** The **attrep_ddl_history** table contains a history of all supported DDL changes that occurred during a task.

  For a list of DDL changes supported by Replicate, see Supported DDL Statements. Note that DDL changes written to this Control Table are also subject to the limitations described in Limitations when Capturing DDL Changes.

  Note Currently, the DDL History table is only supported with the Hadoop target endpoint.

For a detailed description of these tables, see Control Tables.

Bidirectional

This tab is only applicable to bidirectional replication tasks. When you click **Bidirectional** in the **Task Setting**s dialog box, the **Loopback Prevention** tab is displayed. In bidirectional replication, loopback prevention is a mechanism that prevents the same data from being replicated back and forth in an endless loop. To enable loopback prevention, you need to specify a source and target **Loopback prevention table schema**.

Bidirectional replication consists of two separate tasks: Task 1 captures changes made to Endpoint A and replicates them to Endpoint B. Task 2 captures changes made to Endpoint B and replicates them to Endpoint A. When configuring Task 1 of a bidirectional replication setup, the source loopback prevention table schema must be identical to the target loopback prevention table schema specified in the **Loopback Prevention** settings of Task 2.

Likewise, when configuring Task 2 of a bidirectional replication setup, the source loopback prevention table schema must be identical to the target loopback prevention table schema specified in the **Loopback Prevention** settings of Task 1.

Note Oracle schemas are case-sensitive. Therefore, when specifying an Oracle table schema, make sure to use the *correct case* in the **Loopback Prevention** settings in both Tasks.

For instructions on setting up bidirectional replication, see Bidirectional Replication.

Full Load

When you click **Full Load** in the Task Settings dialog box, you can configure the following:

- Full Load Settings
- Full Load Tuning

Full Load Settings

Click the **Full Load Settings** sub-tab to configure the following:

Full is ON/OFF.

Click this button to toggle full load on or off. The initial setting is determined when Setting up Tasks.

When full load is ON, Attunity Replicate loads the initial source data to the target endpoint.

> **Note** Full load can be turned on or off at any stage even if change processing is on. Once the task begins to process changes, the full load on/off switch is used only as additional protection against accidental or unauthorized reload.

Target table preparation:

If target table already exists: Select one of the following from the list to determine how you want to handle loading the target at *full-load start up*:

- **DROP and Create table**: The table is dropped and a new table is created in its place.
- **TRUNCATE before loading**: Data is truncated without affecting the table metadata. Note that when this option is selected, enabling the Create primary key or unique index after full load completes option will have no effect.
- **ARCHIVE and CREATE table:** A copy of the existing table will be saved to the same schema before the new table is created. The archived table name will be appended with a timestamp, indicating when the archiving operation occurred (e.g. `Customers_20170605175601`).

  > **Note** Currently this option is only available for the Hadoop target endpoint.

- **Do nothing**: Existing data and metadata of the target table will not be not affected. New data will be added to the table.

> **Note** Replicate expects the source column data types to be compatible with the corresponding target column data types. If you choose either **TRUNCATE before loading** or **Do nothing** and one or more target data types are different than the data types for the corresponding source columns, use a transformation to convert the data types as required.
>
> For information on creating data type transformations, see Defining Transformations for a Single Table/View.

Create primary key or unique index after full load completes: Select this option if you want to delay primary key or unique index creation on the target until after full load completes.

Stop the task after Full Load completes and: You can set the task to stop automatically after Full Load completes. This is useful if you need to perform DBA operations on the target tables before the task's Apply Changes (i.e. CDC) phase begins.

During Full Load, any changes to the source tables are cached. When Full Load completes, the cached changes are automatically applied to the target tables.

Note This feature is not available for bidirectional replication tasks.

Select **Cached changes have not yet been applied** to stop the task before the cached changes are applied and/or **Cached changes have been applied** to stop the task after the cached changes are applied.

Selecting the **Cached changes have not yet been applied option** will stop the task immediately after Full Load completes. Selecting the **Cached changes have been applied** option will stop the task as soon as data is consistent across all tables in the task.

Note

When configuring Replicate to stop the task after Full Load completes, note the following:

- The task will stop after Full Load completes even if there are no cached changes to apply.
- Choosing to stop the task *before* cached changes have been applied may adversely affect performance, since the cached changes will only be applied to tables (even those that have already completed Full Load) *after* the last table completes Full Load.
- When working with the File Channel endpoint, these options should be set in the remote File Channel task and not in the local File Channel task.

  For more information on the File Channel endpoint, see Using the Attunity Replicate File Channel.

Full Load Tuning

Click the **Full Load Tuning** sub-tab to configure the following:

Tuning settings

- **Maximum number of tables to load in parallel**: Enter the maximum number of tables to load into the target at one time. The default value is **5**.
- **Transaction consistency timeout (seconds)**: Enter the number of seconds that Attunity Replicate waits for transactions to close, if they are open when the task starts, before beginning the Full Load operation. The default value is **600** (10 minutes). Attunity Replicate will begin the full load after the timeout value is reached even if there are open transactions.

  Note: To replicate transactions that were open when Full Load started but were only committed *after* the timeout value was reached, you need to reload the target tables.

- **Commit rate during full load**: The maximum number of events that can be transferred together. The default value is **10000**.

Change Processing

When you click **Change Processing** in the **Task Settings** dialog box, you can configure the following:

- Apply Changes Settings
- Store Changes Settings
- Changes Processing Tuning

Apply Changes Settings

Click the **Apply Changes Settings** sub-tab to configure the following:

Apply Changes is ON/OFF.

Click this button to toggle Apply Changes (Change Processing) on or off. The initial setting is determined when Setting up Tasks.

When Apply Changes is ON, Attunity Replicate processes the changes. You can view the change processing in the Monitor. For more information, see Monitoring Change Processing Operations.

Note When you turn on apply changes you must reload the task or position back to the point of the previous reload.

DDL handling policy: Determine how to handle the target table for the change capture:

- When source table is dropped, select one of the following:
  - **DROP target table**
  - **Ignore Drop**
- When source table is truncated, select one of the following:
  - **TRUNCATE target table**
  - **Ignore TRUNCATE**
- When source table is altered, select one of the following:
  - **ALTER target table**
  - **Ignore ALTER**

Store Changes Settings

When you click **Store Changes** in the Task Settings dialog box, you can configure the **Store Changes Settings** for a replication task.

Store changes processing is ON/OFF

Click this button to toggle **Store Changes** on or off. The initial setting is determined when Setting up Tasks. If this option is ON, changes are stored in either change tables or an audit table.

For more information about storing and applying changes, see Using an Audit Table and Using the Change Table Model.

Note Store Changes can be turned on or off at any time without affecting anything in the task. Changes that are processed and not stored as a result of change storage being turned off can be recovered only by setting the task to an earlier point in time.

If Store Changes is **ON**, use the following options to determine how to store changes. Changes can be stored in Change Tables or in a single Audit table. From the **Store changes in** drop-down list, choose either **Change tables** or **Audit table** according to your needs.

Storing Changes in Change Tables

The following section describes the options that are available when storing changes in Change Tables.

- **Suffix:** Type a string to use as the suffix for all Change Tables. The default value is `__ct`.

  The Change Table names are the name of the target table with the suffix appended. For example, if you have a table called HR and use the default value, the name of the Change Table will be `HR__ct`.

  For more information, see Working with Change Tables.
- **Header column prefix:** Type a string to use as the prefix for all of the Change Table header columns. The default value is `header__`.

  For example, the header column `stream_position` when using the default value is called `header__stream_position`.

  For more information, see Change Tables.
- **DDL options:** Select one of the following options to determine how to handle DDL operations on the source tables:
  - **Apply to change table**: Apply the DDL to the Change Table as well. For example, when this option is enabled and a column is added to one of the source endpoint tables, the column will also be added to the corresponding Change Table.
  - **Ignore**: The change event from any DDL is ignored.
- **On UPDATE:** Select one of the following options to determine how to store UPDATEs to the source tables:
  - **Store before and after image**: To store both the pre-UPDATE data and the post-UPDATE data.
  - **Store after image only**: To store only the post-UPDATE data.

Change table creation:

If change table exists when full load starts: Select one of the following from the list to determine how you want to handle loading the Change Tables at *full-load startup*:

- **DROP and CREATE change table**: The table is dropped and a new table is created in its place.
- **ARCHIVE and CREATE change table:** A copy of the existing table will be saved to the same schema before the new table is created. The archived table name will be appended with a timestamp, indicating when the archiving operation occurred (e.g. `Customers___ct_20170605175601`).

  Note Currently this option is only available for the Hadoop target endpoint.

- **Delete old changes and store new changes in existing change table**: Data is truncated and added without affecting the table metadata.
- **Keep old changes and store new changes in existing change table**: Data and metadata of the existing Change table are not affected.

Change Data Partitioning

Note This feature is supported with the following endpoints only:

- Hadoop target
- File target
- Amazon S3 target

In a standard replication task, changes are replicated to the target in no particular order. Change Data Partitioning enables processing of Change Data from many tables in a consistent fashion. You can define the duration of partitions as well as the partitioning base time, thereby ensuring overall consistency of the partitioned data (i.e. no partial transactions, no order headers without order lines, and so on.)

The partitioned data is stored in the Replicate Change Tables. When the Change Data Partitions table is selected (in the Control Tables tab), information about the partitions will be recorded in the attrep_cdc_partitions Control Table on the target database. This information can be used to identify partitioned data that needs to be further processed.

The partitioning options are as follows:

- **Off** - Replicate Change Data without partitioning.
- **Partition every** - Specify the length (in hours and minutes) of each partition.
- **Partition base time** - Partitions are created during a 24 hour time period, which is calculated according to the specified "Partitioning base time" on the source database (in UTC time). For example, a partition interval of 8 hours with a "Partitioning base time" time of 02:00 will create the following partitions: 02:00-10:00, 10:00-18:00, 18:00-02:00 - but not necessarily in that order. For instance, if a task started at 01:00, then the timeframe of the first partition will be 18:00-02:00. Additionally, if a task started in

the middle of a partition (e.g. at 04:00), its Change Data will be inserted into the 02:00-10:00 partition (even though no changes were captured before 04:00).

Note If there are existing Change Tables that were created before Change Data Partitioning was enabled, you need to drop/rename them so that they can be recreated with the additional "partition_name" column.

Selecting Change Table Header Columns

The Change Table header columns provide information about the Change Processing operation such as the type of operation (e.g. INSERT), the commit time, and so on. If you do not need this information, you can configure Replicate to create the Change Tables without some or all of the header columns, thereby reducing their footprint in the target database. To do this, clear the check boxes next to the header columns that you wish to exclude.

Note that you cannot remove additional columns or restore columns while a task is running. To change your initial selection, you first need to stop the task, then modify your selection, and finally reload the target tables.

Note When Change Data Partitioning is enabled, an extra header column named "partition_name" is added to the Change Tables and automatically selected in the UI. As this column is required, it cannot be excluded.

For a description of the header columns, see Change Tables.

Storing Changes in an Audit Table

The following section describes the options that are available for storing changes in an Audit table.

Note LOB columns with unlimited size are not supported in the CHANGE_RECORD and BU_CHANGE_RECORD fields. The other fields will be recorded but the LOB will have a NULL value.

For a description of the audit table structure, see Using an Audit Table.

- **Audit table schema:** Specify a schema if you do not want the Audit table to be created under the target endpoint's default schema.

  The default schema are as follows:

Endpoint	Default Schema
Pivotal Greenplum	Public
Amazon Redshift	Public
Oracle	The connected user's user name.
Teradata	The endpoint name.
All others	The user's default schema.

- **Audit table tablespace:** This option is only available when the task's target endpoint is Oracle. Enter the tablespace name on the target where you want the Audit table to be created. If you do not enter any information in this field, then the tables will created in the default permanent tablespace.
- **Audit table name:** Specify a name for the Audit table.

  The default value is `attrep__audit_table`.

Audit table creation:

If audit table exists when the target is reloaded: Select one of the following to determine how you want to handle the Audit table when the target is reloaded:

- **DROP and CREATE audit table**: The Audit table is dropped and a new table is created in its place.
- **ARCHIVE and CREATE audit table:** A copy of the existing table will be saved to the same schema before the new table is created. The archived table name will be appended with a timestamp, indicating when the archiving operation occurred (e.g. `attrep_audit_table_20170605175601`).

  Note Currently this option is only available for the Hadoop target endpoint.

- **Delete old changes and store new changes in existing audit table**: Data is truncated and added without affecting the Audit table metadata.
- **Keep old changes and store new changes in existing audit table**: Data and metadata of the existing Audit table are not affected.

For a description of the audit table structure, see Using an Audit Table.

Changes Processing Tuning

Click the **Change Processing Tuning** sub-tab to fine-tune the Apply Changes settings.

Change processing mode

Determine which method will be used to apply changes.

Note Changes to tables without a Unique Index or Primary Key will always be applied in **Transactional apply** mode.

- **Transactional apply**: Select this to apply each transaction individually, in the order it is committed. In this case, strict referential integrity is ensured for all tables.
- **Batch optimized apply**: Select this to commit the changes in batches. In this case, a pre-processing action occurs to group the transactions into batches in the most efficient way. This may affect transactional integrity. Therefore, you must select one of the following to determine how the system will handle referential integrity issues:
  - Preserve transactional integrity

    Note This option is only supported when replicating to an Oracle target.

  - Allow temporary lapses in transactional integrity to improve performance

Note These options are not displayed in bidirectional tasks since such tasks always use the "Preserve transactional integrity" option.

Note The following target endpoints do not support applying binary data types in Batch Optimized Apply mode:

ODBC, SAP Sybase IQ, SAP Sybase ASE, HP Vertica, IBM Netezza, Teradata Endpoint, and Amazon Redshift.

Note When LOB columns are included in the replication, **Batch optimized apply** can only be used with the **Limit LOB size to** option. For more information about including LOB columns in the replication, see Metadata.

Batch tuning

The following options are available when **Batch optimized apply** is selected as the **Change Processing Mode**:

- **Apply batched changes in intervals:**
  - **Longer than**: The minimum amount of time to wait between each application of batch changes. The default value is **1**.

    Increasing the **Longer than** value decreases the frequency with which changes are applied to the target while *increasing* the size of the batches. This can improve performance when applying changes to target endpoints that are optimized for processing large batches, such as Teradata, HP Vertica, and Pivotal Greenplum.
  - **But less than**: The maximum amount of time to wait between each application of batch changes (before declaring a timeout). In other words, the maximum acceptable

latency. The default value is **30**. This value determines the maximum amount of time to wait before applying the changes, after the **Longer than** value has been reached.

- **Force apply a batch when processing memory exceeds (MB)**: The maximum amount of memory to use for pre-processing in **Batch optimized apply mode**. The default value is **500**.

  For maximum batch size, set this value to the highest amount of memory you can allocate to Attunity Replicate. This can improve performance when applying changes to target endpoints that are optimized for processing large batches, such as Teradata, HP Vertica, and Pivotal Greenplum.
- **Limit the number of changes applied per change processing statement to**: To limit the number of changes applied in a single change processing statement, select this check box and then optionally change the default value. The default value is **10,000**.

The following options are available when **Transactional apply** is selected as the **Change Processing Mode:**

- **Minimum number of changes per transaction:** The minimum number of changes to include in each transaction. The default value is **1000**.

  Note Replicate applies the changes to the target either when the number of changes is equal to or greater than the **Minimum number of changes per transaction** value OR when the batch timeout value is reached (see below) - whichever occurs first. Because the frequency of changes applied to the target is controlled by these two parameters, changes to the source records may not immediately be reflected in the target records.

- **Maximum time to batch transactions before applying (seconds):** The maximum time to collect transactions in batches before declaring a timeout. The default value is **60**.

Transaction offload tuning

The following tuning options are available, regardless of which **Change processing mode** is selected:

- **Offload transaction in progress to disk if**:

  Attunity Replicate usually keeps transaction data in memory until it is fully committed to the source and/or target. However, transactions that are larger than the allocated memory or that are not committed within the specified time limit will be offloaded to disk.

  - **Transaction memory size exceeds (MB)**: The maximum size that all transactions can occupy in memory before being offloaded to disk. The default value is **1000**.
  - **Transaction duration exceeds (seconds)**: The maximum time that each transaction can stay in memory before being offloaded to disk. The duration is calculated

from the time that Attunity Replicate started capturing the transaction. The default value is **60**.

Miscellaneous tuning

- **Statements cache size (number of statements):** The maximum number of prepared statements to store on the server for later execution (when applying changes to the target). The default is 50. The maximum is 200.
- **Store task recovery data in target database:** Select this option to store task-specific recovery information in the target database. When this option is selected, Replicate creates a table named `attrep_txn_state` in the target database. This table contains transaction data that can be used to recover a task in the event that the files in the **Data** folder are corrupted or if the storage device containing the **Data** folder has failed.

  For more information about this option, see Recovering from Data Folder Loss or Corruption.

Error Handling

When you click **Error Handling** in the Task Settings dialog box, you can configure the following:

- Error Handling Settings
- Environmental Errors
- Data Errors
- Table Errors
- Apply Conflicts

For more information on error handling in Attunity Replicate, see Error and Crash Handling.

Error Handling Settings

The option to switch between the Global Error Handling policy and a Task-Specific Error Handling policy is available in each of the **Error Handling** sub-tabs. However, the policy you enable will be applied to all error types, regardless of where it was enabled. For example, you cannot enable a Task-Specific Error Handling policy for Data Errors and then enable the Global Error Handling policy for Table Errors and Environmental Errors.

For information on setting the global error handling policy, see Global Error Handling.

To set a Task-Specific Error Handling policy:

- Click the **Change to Task Specific Policy** button in any of the **Error Handling** sub-tabs.

To revert to the Global Error Handling policy:

1. Click the **Change to Global Policy** button in any of the **Error Handling** sub-tabs.
2. Click **OK** when prompted to confirm your action.

Environmental Errors

Click the **Environmental Errors** sub-tab to configure the following:

- **Maximum retry count**: Select this option and then specify the maximum number of attempts to restart a task when an environmental error occurs.

  Specify "0" to never restart a task.

  When the check box is not selected, Attunity Replicate will attempt to restart the task an infinite number of times.

  When the system attempts to restart the task the designated number of times, the task is stopped and manual intervention is required.
- **Interval between retry attempts**: Use the counter to select or type the number of seconds that the system waits between attempts to restart a task.
- **Increase retry interval for long outages**: Select this check box to increase the retry interval for long outages. When this option is enabled, the number of seconds between retry attempts increases each time.
- **Maximum retry interval**: Use the counter to select or type the number of seconds to wait between attempts to restart a task when the **Increase retry interval for long outages** option is enabled.

For information about environmental errors and the configuration properties, see Environmental Errors and Error Handling Properties in the Error and Crash Handling appendix.

Data Errors

Click the **Data Error** sub-tab to configure the following:

- **For a data truncation error:** Click the triangle to open the list and select what happens when an truncation occurs in one or more specific records. You can select one of the following from the list:
  - **Ignore record**: The task continues and the error is ignored.
  - **Log record to the exceptions table** (default): The task continues and the error is written to the exceptions table.
  - **Suspend table**: The task continues but data from the table with the error record is moved into an error state and its data is not replicated
  - **Stop task**: The task is stopped and manual intervention is required.

- **For other data errors:** Click the triangle to open the list and select what happens when an error occurs in one or more specific records. You can select one of the following from the list:

- **Ignore record**: The task continues and the error is ignored.
- **Log record to the exceptions table** (default): The task continues and the error is written to the exceptions table.
- **Suspend table**: The task continues but data from the table with the error record is moved into an error state and its data is not replicated
- **Stop task**: The task is stopped and manual intervention is required.

- **Escalate error handling when other data errors reach (per table)**: Select this check box to escalate error handling when the number of non-truncation data errors (per table) reaches the specified amount.
  - **Escalation action**: Choose what action Replicate should perform when error handling is escalated. Note that the available actions are dependent on the action selected from the **For other data errors** drop-down list described above.
    - **Log record to the exceptions table:** The task continues and the error is written to the exceptions table.
    - **Suspend table** (default): The task continues but data from the table with the error record is moved into an error state and its data is not replicated.
    - **Stop task**: The task is stopped and manual intervention is required.

For information about environmental errors and the configuration properties, see Data Errors and Error Handling Properties in the Error and Crash Handling appendix.

Table Errors

Click the **Table Errors** sub-tab to configure the following:

- **When encountering a table error**: Select one of the following from the drop-down list:
  - **Suspend table** (default): The task continues but data from the table with the error record is moved into an error state and its data is not replicated
  - **Stop task**: The task is stopped and manual intervention is required.
- **Escalate error handling when table errors reach (per table)**: Select this check box to escalate error handling when the number of table errors (per table) reaches the specified amount.
  - **Escalation action**: The escalation policy for table errors is set to **Stop task** and cannot be changed.

For information about environmental errors and the configuration properties, see Table Errors and Error Handling Properties in the Error and Crash Handling appendix.

Apply Conflicts

Click the **Apply Conflicts** sub-tab to configure the following:

- **No record found for applying a DELETE**: Click the triangle to open the list and select what happens when there is a conflict with a DELETE operation. You can select one of the following from the list:

- **Ignore record** (default): The task continues and the error is ignored.
- **Log record to the exceptions table**: The task continues and the record is written to the exceptions table.
- **Suspend table**: The task continues but data from the table with the error record is moved into an error state and its data is not replicated
- **Stop task**: The task is stopped and manual intervention is required.

- **Duplicate key when applying an INSERT**: Click the triangle to open the list and select what happens when there is a conflict with an INSERT operation. You can select one of the following from the list:
  - **Ignore record**: The task continues and the error is ignored.
  - **Log record to the exceptions table** (default): The task continues and the record is written to the exceptions table.
  - **Suspend table**: The task continues but data from the table with the error record is moved into an error state and its data is not replicated
  - **Stop task**: The task is stopped and manual intervention is required.
  - **Update the existing target record**: The target record with the same primary key as the INSERTED source record is updated.
- **No record found for applying an UPDATE**: Click the triangle to open the list and select what happens when there is a conflict with an UPDATE operation. You can select one of the following from the list:
  - **Ignore record**: The task continues and the error is ignored.
  - **Log record to the exceptions table** (default): The task continues and the record is written to the exceptions table.
  - **Suspend table**: The task continues but data from the table with the error record is moved into an error state and its data is not replicated
  - **Stop task**: The task is stopped and manual intervention is required.
  - **Insert the missing target record**: The missing target record will be inserted into the target table. When the source endpoint is Oracle, selecting this option requires supplemental logging to be enabled for all the source table columns.

    Note When this option is selected, LOB columns in the source tables will not be replicated to the target.

- **Escalate handling when apply conflicts reach (per table)**: Select this check box to escalate error handling when the number of apply conflicts (per table) reaches the specified amount.
- **Escalation action**: Choose what action Replicate should perform when handling is escalated. Note that the available actions are dependent on the action selected in the drop-down lists described above.

- **Log error** (default): The task continues and the error is written to the task log.
- **Suspend table**: The task continues but data from the table with the error record is moved into an error state and its data is not replicated
- **Stop task**: The task is stopped and manual intervention is required.

For information about environmental errors and the configuration properties, see Apply Errors and Error Handling Properties in the Error and Crash Handling appendix.

Note When you select **Fix record** you must be sure that you are using full supplemental logging to ensure that an UPDATE is not turned into an INSERT. In other cases, FIX_RECORD can cause an async full load of a record similar to the LOB channel.

Logging

You can set the logging level for task logs by selecting the **Logging** tab in the **Task Settings** dialog box and then selecting the **Logging Level** sub-tab.The level you set determines what information is written to the log

Note You can also set the task logging level from the **Tools** menu in the Monitor view.

For more information, see Monitor Mode and Setting the Task Logging Level.

The following are the available logging levels. The list is in order from the lowest level to the highest level.

1. Error
2. Warning
3. Info
4. Trace
5. Verbose

The higher levels always include the messages from the lower levels. Therefore, if you select **Error**, only error messages are written to the log. However, if you select **Info**, informational messages, warnings, and error messages are included. Selecting **Verbose** writes all possible messages to the log.

For information on how to set the logging level, see Setting the Task Logging Level.

12 | Working with Tasks at Runtime

This section describes how to work with tasks that you design. For information on how to design a task, see Designing Tasks. This chapter contains information on running tasks, viewing the task status, and viewing messages about the task. Information on monitoring and working with tasks during runtime is in the section Monitoring and Controlling Replication Tasks.

In this chapter:

Running a Task

After you design a task (see Designing Tasks), you can run and monitor its progress with one click in Attunity Replicate. This simple Click-2-Replicate function is described in this topic. In addition, the various types of run options available are also described. This topic has the following sub-topics.

- How to Run a Task
- Using the Run Button Options

Note The task run buttons area available in the toolbar at the top of the console in the following views:

- Tasks View (in both Designer Mode and Monitor Mode)
- When Viewing Specific Tasks

How to Run a Task

Click the **Run** button to execute a replication task. The task process continues to run until you click the **Stop** button to stop the task.

Note When you click **Run**, the following occurs:

- If this is the first time that a task is run, the Start Processing operation is run.
- If the task has been started and stopped, the **Resume Processing** operation described in Using Advanced Run Options is run.
- If changes were made to the endpoint, change processing takes place after the full load operation. If you do not want change processing to occur or if you want to start change processing from a predetermined point, you must make the appropriate Using Advanced Run Options selection.

In some cases, task replication may stop due to an error although the task process is still running.

See Tasks View for information on the task status and how Attunity Replicate displays information on the current task status.

The **Run** button is available in the following views:

- The Tasks view when you select a task from the Task List.
- For the individual task, both the Designer mode and Monitor mode have the **Run** and **Stop** buttons available.

Note You must be in the Monitor mode to view the task progress.

Using the Run Button Options

Clicking the **Run** button runs a full-load replication task from the source to the target. This is a first time task that creates the target endpoints and loads the source data to the target according to your task definitions.

Subsequent runs allow you to resume processing from a specific point and process changes. In addition, you can also specify from what point you want the replication to start.

The following options are available:

- Start Processing (switches to **Resume Processing** after the task has started)
- **Resume Processing:** Resumes task execution from the point that it was stopped. You can also resume processing by clicking the **Run** button if the task has been stopped.

  Note If the schema or a filter was changed after the task stopped, the task should be reloaded as opposed to resumed (see below).

- Reload Target (Only available when the **Full Load** or **Full Load** *and* **Apply Changes** replication options are enabled)
- Using Advanced Run Options

Start Processing

This is available the first time you run the task only. This will execute the initial full load operation. If Change Processing is also enabled for the task or if it is an Apply Changes only task type, change processing will start as soon as any changes are made to the source endpoint.

Reload Target

Starts the full load and change processing (if enabled) from the beginning. Already processed tables are handled according to the Task Setting, **Initial Table Creation; If target table already exists**. See Full Load Tuning for more information about this setting.

Note To replicate tables that were added to the local file channel task after the initial full load, you need to reload both the local and the remote file channel tasks.

Using Advanced Run Options

Advanced Run Options provide you with additional options for resuming and restarting tasks.

To use Advanced Run Options, click the triangle next to the **Run** button and select **Advanced Run Options**.

The **Advanced Run Options** dialog box opens.

The **Advanced Run Options** dialog box lets you do the following:

- ****Restart task and start processing changes from current time**: This starts the Apply Changes replication task from the beginning (as if the task has not run before).

  **Only available for Apply Changes replication tasks.
- **Tables are already loaded. Start processing changes from**:

  Note When resuming a task from MySQL, the **Date and Time** or **Source change position** must always correspond to the beginning of a transaction.

  - **Date and Time:** Select the date and time from where you want to Replicate to start processing changes.

    Notes

    - When logs are deleted from the database (e.g. due to a purge policy), a log matching the specified date and time may not exist. In this case, Replicate will resume the task from the earliest point it can after the specified date and time.
    - The timestamp uses the local time of the browser machine.
    - This option is not relevant for the File Source endpoint.

- **Source change position (e.g. SCN or LSN):** Specify the position in the log from where to resume change processing. The source change position format differs according to your source endpoint. For instance, to resume processing from a Microsoft SQL Server database, you would need to specify the LSN (e.g. `000000c1:00001d6f:0004`). However, if your source endpoint is Oracle, you would need to specify the SCN (e.g. `10737419121`).

  Note The Source change position option is supported with the following source endpoints only:

  Oracle, Microsoft SQL server, MySQL and PostgreSQL.

Metadata Only:

The "Metadata only" options described below allow you to:

- Create empty tables on the target and then manually edit them.
- Create tables during a task.

Enabling the options will also ensure that supplemental logging is set up correctly on the source tables before starting the actual replication task.

- **Recreate all tables and stop:** Select this option to recreate the target tables as defined in the **Full Load Settings** tab. When "Store Changes" is enabled, the Change tables/The Audit table will be created as defined in the **Store Changes Settings** tab. To use this option, stop the existing task, run the task with this option enabled (the task will stop automatically) and finally, resume the task.
- **Create missing tables and stop:** Select this option to create missing target tables including Change Tables. You can use this option to create Change Tables on the target after enabling the "Store Changes" option (in the **Store Changes Settings** tab) for an existing task. To use this option, stop the existing task, run the task with this option enabled (the task will stop automatically) and finally, resume the task.

Recovery:

- **Recover using locally stored checkpoint:** Use this option if recovery is not possible using the **Resume Processing** or **Start process changes from** options (due to corrupt swap files, for example). When this option is selected, Replicate uses the checkpoint data stored in `<Data_Folder_Path>\data\tasks\<task_name>\StateManager` to recover the task.

  Note When using this option, the following limitations apply:
  - The following source endpoints are supported only:
    - Oracle
    - Microsoft SQL Server
  - Tasks can only be recovered during Change Processing (i.e. after Full Load Completes)

- With the exception of the File Channel endpoint, all target endpoints are supported. The following limitations apply:
  - **In Transactional apply Change Processing mode:** All target endpoints that support transactions are supported.
  - **In Batch optimized apply Change Processing mode:** Oracle target endpoint only is supported. Also requires the **Preserver transactional integrity** option to be enabled.
  - For all other target endpoints or Change Processing modes, recovery is supported, but may cause duplicates on the target.

- **Recover using checkpoint stored on target:** Select to recover a task using the `CHECKPOINT` value from the `attrep_txn_state` table (created in the target database).

  Note When using this option, the following limitations apply:
  - Only the following source and target endpoints are supported:
    - Oracle
    - Microsoft SQL Server
  - Tasks can only be recovered during Change Processing (i.e. after Full Load Completes)
  - The task Change Processing mode must be set to either*:*

    Batch optimized apply with the **Preserver transactional integrity** option enabled. Note that setting this mode may cause duplicates on Microsoft SQL Server target.

    -OR-

    Transactional apply

    For information about setting the Change Processing mode, see Changes Processing Tuning.

  This option will only be available if the **Store task recovery data in target database** option was enabled in the Task Settings' Changes Processing Tuning tab *before* Change Processing completed.

  Select this option (as opposed to the **Recover using locally stored checkpoint** option) if the files in the **Data** folder are corrupted or if the storage device containing the **Data** folder has failed.

  For a detailed explanation of how to set up and implement recovery using the `attrep_txn_state` table, see Recovering from Data Folder Loss or Corruption.

Recovering from Data Folder Loss or Corruption

During normal operation, Attunity Replicate maintains the replication state in the following location:

`<Data_Folder_Path>\data\tasks\<task_name>\StateManager`

This enables tasks that cannot be resumed normally (due to corrupt swap files, for example) to be recovered using the **Recover using locally stored checkpoint** option described in Using Advanced Run Options.

However, if the files in the **data** folder become corrupted or if the storage device containing the **data** folder fails, tasks must be recovered using the means described below.

Setting Up and Initiating Task Recovery

For recovery to be successful, the source database transaction logs must be available from the time the task failed.

To set up a task for recovery

1. Design a task. Make sure to enable the **Store task recovery data in target database** option in the Task Settings' Changes Processing Tuning tab. This option can be enabled at any time during Change Processing, although it must be enabled *before* Change Processing completes.
2. Export the task definitions as described Exporting Tasks.
3. Run the task.

In addition to the selected source tables, the task will write the checkpoint data to the following table in the target database (and automatically create the table if it has not already been created by another task):

`attrep_txn_state`

To initiate recovery

1. Import the task definition exported when you set up the task.
2. Enter the passwords in the endpoint connection settings.
3. Access the `attrep_txn_state` table on the target database and locate the failed task in the `TASK_NAME` column. If there are tasks with the same name running on multiple Replicate Servers, you will also need to locate the appropriate server in the `SERVER_NAME` column. After locating the relevant task, copy the value in the corresponding `CHECKPOINT` column.
4. Select the **Recover using checkpoint stored on target** option and then provide the `CHECKPOINT` value (preferably by pasting) as described in Using Advanced Run Options.
5. Click **OK** to start the recovery.

During recovery, Replicate does not write anything to the target database until it identifies the commit event corresponding to the `CHECKPOINT` value. Once it identifies the `CHECKPOINT` commit event, recovery is performed and the task reverts to standard operation.

Viewing the Task Status

In the Tasks View, you can see the task status by viewing the icon for the task. After a task is run, the task icons in the Task view display the current status of the task. For additional information on the possible statuses, see Tasks View.

The following icon represents a task that is in an error status.

There are two types of errors:

Recoverable error: A recoverable error indicates that there is a temporary problem, such as a missing connection. The task icon is blue indicating that the task is still active. In this case, Attunity Replicate attempts to restart the task automatically. As soon as the error state is resolved, the task is restarted.

The task remains active but paused throughout the error state. You can stop the task at any time and resolve the error manually, if necessary.

Note Attunity Replicate will continue to check the task for 30 minutes to determine whether it is no longer in an error status. If the error is not resolved in 30 minutes the error becomes a fatal error and you must resolve the error manually.

Fatal Error: When a fatal error occurs, the task stops and you must resolve the error manually. You cannot start the task again until the error is resolved. Use the logs or the messages in the Alerts pane to see the error type.

See also:

View Log Messages for a Task

Viewing Notifications

Reading Messages about a Task

Task messages are displayed in the **Messages** section of the Attunity Replicate Console. The **Messages** section is located at the bottom right of the console in the Monitor Mode and when Viewing Specific Tasks.

The Message section has two types of messages that provide information about events that occur in a task. Each type of message is displayed in the following tabs:

- Viewing Notifications
- View Log Messages for a Task

Viewing Notifications

The **Notifications** tab displays notifications about the task. These messages alert you to specific events encountered by a task, such as the task starting or stopping, a specific error type, or information about latency and disk space.

The **Notifications** tab displays the time of a notification and a description of the notification. You define the notifications that are sent for each task and a description for each notification in the Settings area. For more information, see Define the Notification Message.

Using the Notifications List

When a notification is sent, it is displayed in the **Notifications** tab. This section describes the tasks that can be performed in the **Notifications** tab.

Opening a Notification

When you open a notification, you can see the full message presented in a dialog box. The dialog box contains a button to copy the text so that you can use it somewhere else for troubleshooting and the timestamp for the notification.

To open a notification:

1. In the Messages section of the console, click the **Notifications** tab. The **Notifications** tab opens.
2. Select the notification you want to open from the list.
3. Double-click the notification or click **Open** from the toolbar at the top of the list.

Clearing a Notification

You can clear notifications from the list to make sure that you are seeing only those that are relevant to you.

To clear a notification:

1. In the Messages section of the console, click the **Notifications** tab.
2. Select the notification you want to clear from the list.
3. Click **Clear** from the toolbar at the top of the list.

Sorting Notifications

You can sort log messages according to Date and Time and Message.

To sort the notifications:

1. In the **Messages** section of the console, click the **Notifications** tab.
2. Click the **Date and Time** or **Message** column according to how you want to sort the messages.

An upward arrow indicates that the column is sorted in ascending order whereas a downward arrow indicates that the column is sorted in descending order.

View Log Messages for a Task

The **Log Messages** tab displays log messages for errors or warnings from a task. The errors are listed in this tab with the time of the error or warning and the log entry for the event. You can choose to view both errors and warnings or only one of them.

If errors or warnings exist in the task, a red circle with the total number of errors and warnings is displayed. The number displayed may be the number of errors, the number of warnings, or the total of number of errors and warnings depending on what you select to view in this tab. The **Log Messages** tab is shown in the figure below.

Using the Log Messages List

When a log error or warning is sent, it is displayed in the **Log Messages** tab. This section describes the tasks that can be performed in the **Log Messages** tab.

Selecting the Log Message Type

Two types of log messages are displayed in the Log Messages List. You can view Errors, Warnings, or both.

To select the log message type:

- Select the check box or boxes for the type messages you want to view. The check boxes are located at the top of the Log Messages List.

Opening a Log Message

When you open a log message, you can see the full log text presented in a dialog box. The dialog box contains a button to copy the text so that you can use it somewhere else for trouble shooting and the timestamp for the log message.

To open a log message:

1. In the **Messages** section of the console, click the **Log Messages** tab.
2. Select the log message you want to open from the list.
3. Double-click the log message or click **Open** from the toolbar at the top of the list.

Clearing a Log Message

You can clear log messages from the list to make sure that you are seeing only those that are relevant to you.

To clear a log message:

1. In the Messages section of the console, click the **Log Messages** tab.
2. Select the log message you want to clear from the list.

3. Click **Clear** from the toolbar at the top of the list.

Sorting Log Messages

You can sort log messages according to Date and Time, Level and Message.

To sort the log messages:

1. In the Messages section of the console, click the **Log Messages** tab.
2. Click the **Date and Time**, **Level** or **Message** column according to how you want to sort the messages.

   An upward arrow indicates that the column is sorted in ascending order whereas a downward arrow indicates that the column is sorted in descending order.

Viewing the Log file in the Log Viewer

In addition to viewing the log messages, you can view the entire log file in the log viewer.

To view the log in the log viewer:

» From the **Messages** section, click **View Logs**.

The Log Viewer opens.

For a description of actions you can perform in the Log Viewer, see Viewing the Task Log Files and Manually Rolling them Over .

13 | Monitoring and Controlling Replication Tasks

When you monitor and run a task, you can use the Click-2-Replicate function to carry out the replication task and view its functions in near real time. This section describes how to run and monitor a replication task.

In this chapter:

Viewing Information in the Monitor

You access the Monitor view when you open a specific task. The monitor provides near real-time information for the task you select.

To access the Monitor:

1. When Viewing Specific Tasks, select the task you want to monitor.
2. From the toolbar at the top of the console, click **Open**.
3. From the toolbar at the top right, click **Monitor**.

   The Monitor opens. To view the information in real time, you need to run the task (if the task has not already started). For information on running a task, see Running a Task.

Monitoring Full-Load Operations

You can view the progress of a full-load operation in the left side of the Monitor.

To make sure you are viewing the information for a full-load operation, select the **Full Load** tab.

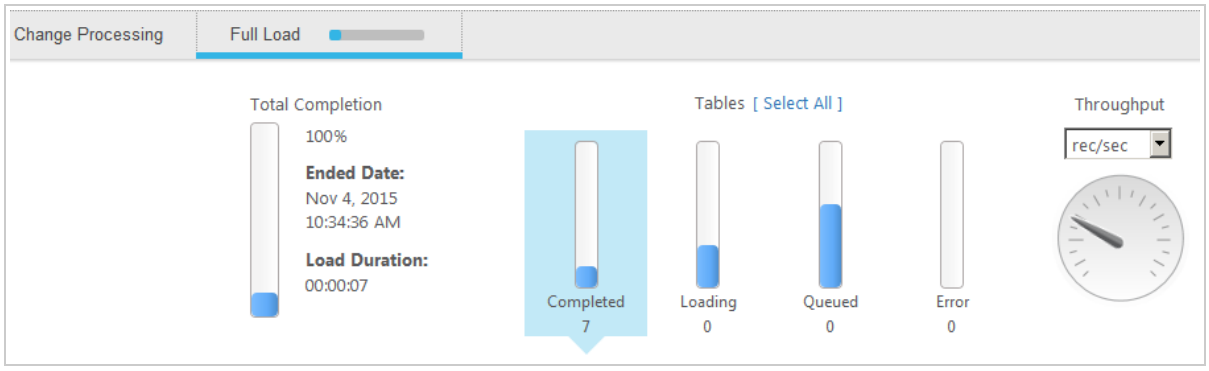
You can view the following:

- General Information for a Full Load
- Detailed Information for the Full Load
- Monitoring Throughput in a Full Load Operation

General Information for a Full Load

General information about the full load is presented in a graphical format. The following figure shows the graphical information displayed when a task is running.

Figure 13.1 | Full Load Status Bars



This section has the following information:

- Status bars: Indicates the status of the tables being loaded.
  - **Completed**: The number of tables that finished loading into the target endpoint.
  - **Loading**: The number of tables that are in the process of loading into the target endpoint.
  - **Queued**: The number of tables that are waiting to load into the target endpoint.
  - **Error**: The number of tables that could not be loaded due to an error. See Reading Messages about a Task for information about error messages.
- Full-load total completion bar: Displays the progress of all records being loaded to the target endpoint. The bar is located in the **Full Load** tab at the top of the graph section.
- Throughput gauge: Displays the current throughput.Throughput displays the number of events read in the task for a specified amount of time.

You can also view Detailed Information for the Full Load.

Detailed Information for the Full Load

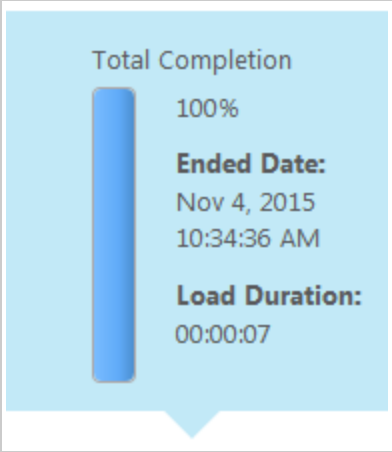
For each of the status bars displayed in the General Information for a Full Load graphs, a table is displayed in the section below with specific information about the current loading status. The following information is available:

- General Information for a Completed Task
- Information for Each Table in the Task
- Information for Tables that have Completed Loading
- Information for Tables that are Currently Loading
- Information for Tables that are in the Loading Queue
- Information for Tables with Errors

General Information for a Completed Task

This section displays a table with information for all of the completed tables in a task. To view this table, click the **Total Completion** bar, shown in the figure below.

Figure 13.2 | Total Completion Status



This table displays the following Progress Details:

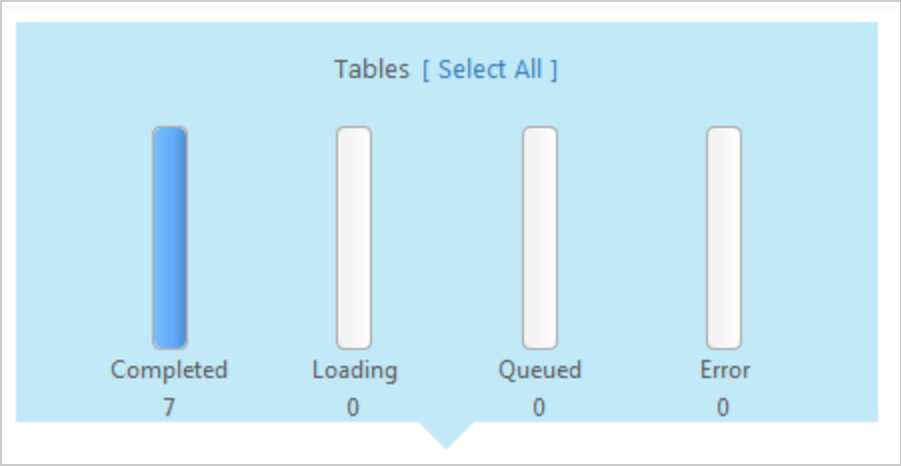
Table 13.1 | Progress Details for all Tables in the Task

	Total	Completed	Remaining	Notes
Tables	The total number of tables that are included in the task.	The total number of tables that completed loading at the current time.	The total number of tables waiting to be loaded.	Additional information.
Records	The total records that completed loading at the current time.	The total number of records that completed loading at the current time.	The total number of records waiting to be loaded.	Additional information.
Time	The estimated time to load all of the selected tables in the task.	The total elapsed time.	The estimated amount of time to load the remaining tables.	Additional information.

Information for Each Table in the Task

This section describes the progress of each of the tables being processed for the task. To display this information, click the **[Select all]** link above the

Figure 13.3 | Select All Tables



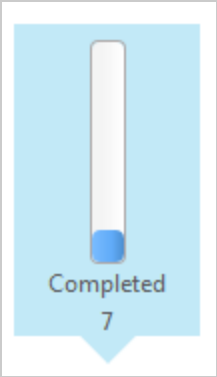
The information is displayed in a table that has the following columns:

- **Table Name**: The names of the source tables that are included in the task.
- **Status**: This is a statement that describes the status for the table. The following are the statuses that can be displayed:
  - **Queued**: The table is in the queue waiting to be loaded to the target endpoint.
  - **Loading**: The table is being processed but is not finished loading.
  - **Completed**: All of the table records are loaded to the target.
  - **Error**: The table stopped loading due to an error. See Reading Messages about a Task for more information about task errors.
- **Estimated Count**: The number of records that are loaded to the target.
- **Elapsed Time**: The total elapsed time since the table records began processing.
- **Progress**: The table status and the time the table entered that status.
- **Reload**: To reload selected tables, select the tables you want to reload and then click the **Reload** button above the table list. When prompted to confirm the operation, click **OK**. The data in the selected tables will be reloaded to the target endpoint. Note that this option is not available for **Apply Changes Only** tasks.

Information for Tables that have Completed Loading

This section displays a table with information for each of the completed tables. To view this table, click the **Completed** bar, shown in the figure below.

Figure 13.4 | Completed Tables Status



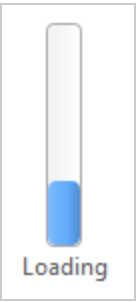
The information is displayed in a table that has the following columns:

- **Table name**: The names of the source tables that have completed loading.
- **Loaded On**: The time that the table completed loading all of its records to the target.
- **Transferred Count**: The number of records loaded to the target.
- **Transferred Volume**: The volume of the records (in KB) loaded to the target.
- **Load Duration**: The amount of time that it took for all records to load to the target.
- **Throughput Records**: The average throughput rate for the table. Throughput describes the number of records read per second. For more information on throughput, see Monitoring Throughput in a Full Load Operation.
- **Throughput Volume**: The average throughput rate for the table. Throughput describes the volume of records (in KB) read per second. For more information on throughput, see Monitoring Throughput in a Full Load Operation.
- **Reload**: Click the **Reload** icon to reload the data for selected tables and run the full-load operation again.

Information for Tables that are Currently Loading

This section displays a table with information for each of the tables that are currently loading. To view this table, click the **Loading** bar, shown in the figure below.

Figure 13.5 | Loading Tables Status



The information is displayed in a table that has the following columns:

- **Table Name**: The names of the source tables that are currently loading.
- **Load Duration**: The amount of time that it took for all records to load to the current point in time.
- **Estimated Count**: The estimated number of rows that are to be loaded in the full load operation.
- **Transferred Count**: The number of records that are loaded to the target endpoint.
- **Current Throughput**: The current throughput rate for the table. Throughput describes the number of records read per second. For more information on throughput, see Monitoring Throughput in a Full Load Operation.
- **Estimated Finish Time**: The approximate time the task finished loading the tables. The timestamp displayed indicates the date and time.

  Note There may sometimes be a discrepancy between the "Estimated Finish Time" and the"Time Remaining (Estimated)" values.

  The "Time Remaining (Estimated)" value is calculated by the combined transfer rate of all the records of the task, while the "Estimated Finish Time" is calculated per table.

  The discrepancy arises when the table transfer rate at the beginning of the task is very fast, but slows down towards the end of the task.

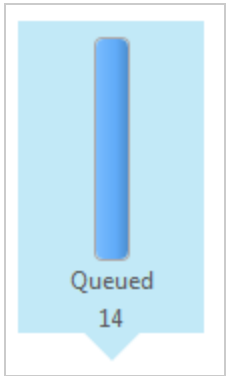
  In this situation, the "Time Remaining (Estimated)" value will be greater and less accurate than the "Estimated Finish Time" value.

- **Progress**: The table status and the time the table entered that status.
- **Reload**: Click the **Reload** icon to reload the data for selected tables and run the full-load operation again.

Information for Tables that are in the Loading Queue

This section displays a table with information for each of the tables that are waiting to be loaded. To view this table, click the **Queued** bar, shown in the figure below.

Figure 13.6 | Queued Tables Status



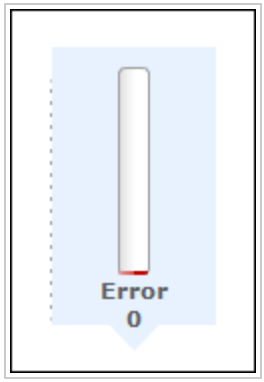
The information is displayed in a table that has the following columns:

- **Table Name**: The names of the source tables that are currently in the queue waiting to be loaded.
- **Estimated Count**: The estimated number of rows that are waiting to be loaded in the full load operation.

Information for Tables with Errors

This section displays a table with information for each of the tables that stopped loading or suspended CDC due to an error. To view this table, click the Error bar, shown in the figure below.

Figure 13.7 | Error Tables Status



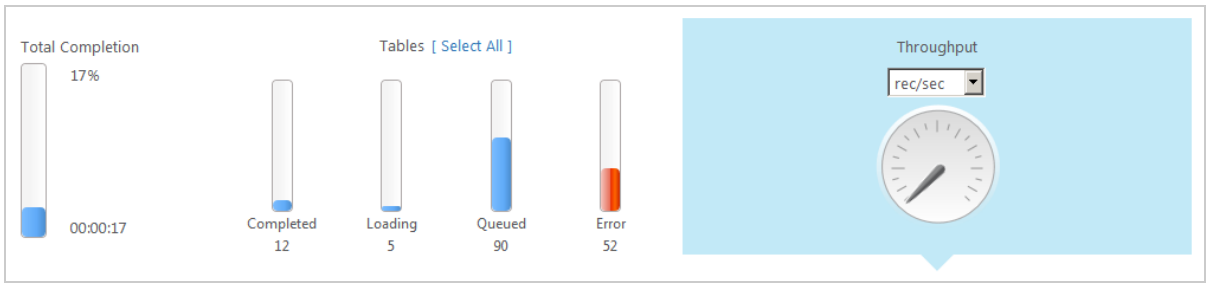
The information is displayed in a table that has the following columns:

- **Table Name**: The names of the source tables that stopped due to an error.
- **Failed On**: The time that the error occurred.
- **Loaded Count**: The number of records loaded when the error occurred.

Monitoring Throughput in a Full Load Operation

Throughput values for a full-load operation provide information on how fast the table records are being replicated to the target endpoint. The information is displayed in a gauge on the right side of the full-load graph section. The following figure shows the throughput gauge.

Figure 13.8 | Throughput Gauge



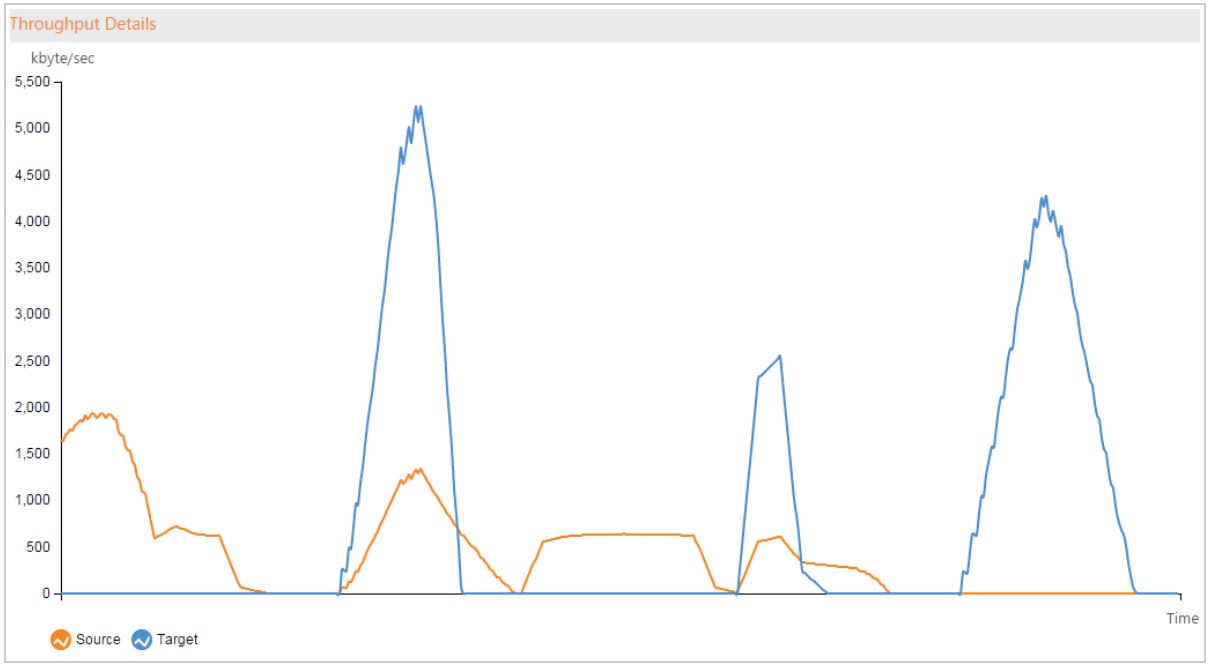
You can set the throughput measurement values either to the number of records replicated per second, or to the number of kilobytes replicated per second. The display is always based on the current load operation.

To set the unit of throughput measurement:

- Select either **rec/sec** or **kbyte/sec** from the drop-down menu above the Throughput gauge.

Click the **Throughput** gauge to display a graph with the throughput details as shown in the figure below. To view the graph only, click the expand/collapse arrow in right side of the gray bar above the graph. Click the arrow again to restore the status bars and throughput gauge.

Figure 13.9 | Throughput Details



Monitoring Change Processing Operations

You can view the progress of the change-processing operation in the left section of the Monitor.

To make sure you are viewing the information for a change-processing operation, select the **Change Processing** tab.

You can view the following:

- General Change Processing Information
- Detailed Change Processing Information

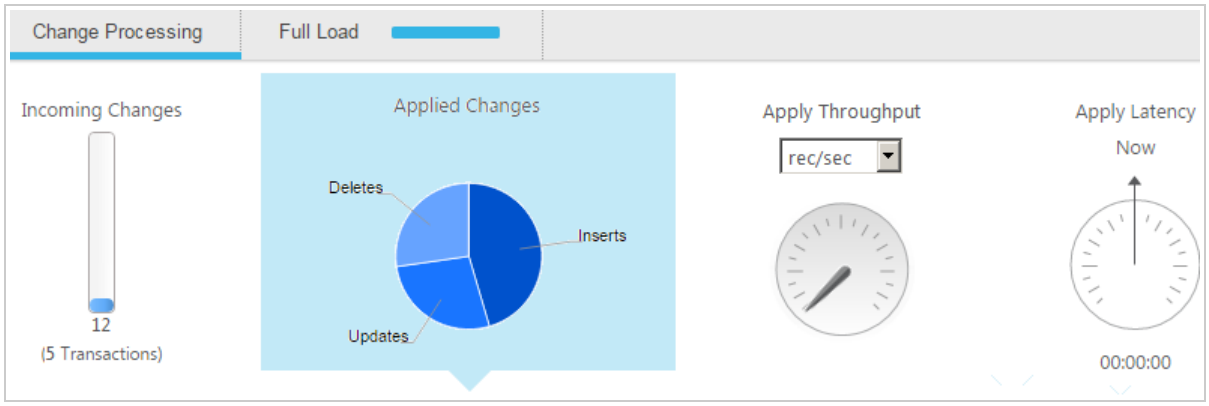
General Change Processing Information

General information about the Change Processing task is presented in a graphical format. The following figure shows the graphical information displayed.

Note In an Apply Changes only task, Replicate performs the following operations depending on whether or not the target tables already exist:

- If a target table does not exist, Replicate will create it (metadata only).
- After the table is created, only INSERT operations will be supported. Other operations (e.g. UPDATE) will fail since the data does not exist (unless it has been inserted earlier).
- If the table already exists, Replicate will behave according to the If target table already exists setting in the task settings' **Full Load Settings** tab.

Figure 13.10 | Change Processing Status



This section has the following information:

- **Incoming Changes**: The total number of records that were processed for the task.
- **Applied Changes**: A circle graph that shows information about the processed changes. It displays the following:
  - The number of INSERT operations processed. Roll over the Insert section with your mouse to see the number and percentage of the accumulated inserts.
  - The number of UPDATE operations processed. Roll over the Update section with your mouse to see the number and percentage of the accumulated updates.
  - The number of DELETE operations processed. Roll over the Delete section with your mouse to see the number and percentage of the accumulated deletes.
  - The number of metadata changes (DDL) processed. DDL changes include information about events like changes to table names or to column names.
- **Apply Throughput** gauge: A gauge that describes the number of change events read per second. For additional details, you can also view a graph with Information about Change Processing Throughput.
- **Apply Latency** gauge: A gauge that displays the latency information.

The latency values displayed in the Attunity Replicate Console measure the time delay (latency) between the time when a change is visible to the source (and committed), and the time when this same change is visible to the target. The display is always based on the current change being applied.

You should take the following into consideration:

- Latency when applying large transactions:

  For example, when the most recent latency value was 10 seconds and now a transaction of one million rows gets committed at the source endpoint, Attunity Replicate starts to apply that transaction to the selected target and it will take some time to write all the changes to the target (for example 60 seconds). During the next 60 seconds, the latency value gradually grows to 70 seconds for the last change in the transaction. Once the transaction is committed, the latency drops back to the 'regular' latency (10 seconds in this case).

- Latency when no transactions are being applied:

  When a time period passes with no changes applied to the target, the latency calculation is based on the time difference between the current time and the timestamp of the last change event read from the transaction log. This could happen if, for example, there is high activity on tables which are not selected for replication in the current task.

For additional details, you can also view a graph with Information about Apply Latency.

Detailed Change Processing Information

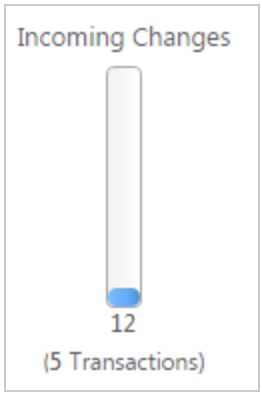
For each of the status indicators displayed in the General Change Processing Information section, a table or graph is displayed in the section below with detailed information about the change processing status. The following information is available:

- Information about Incoming Changes
- Information about Applied Changes
- Information about Change Processing Throughput
- Information about Apply Latency

Information about Incoming Changes

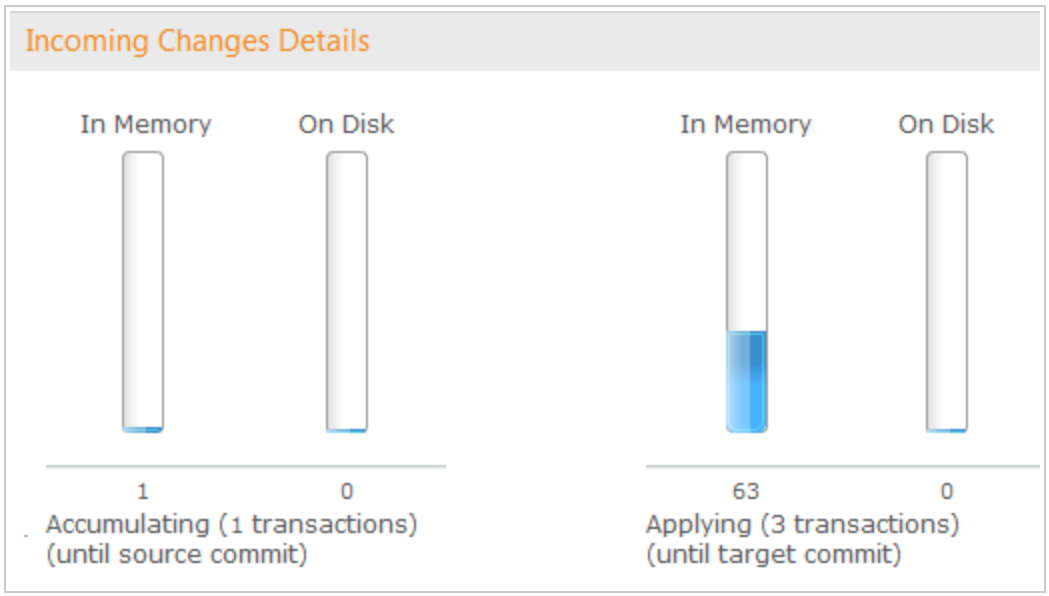
This section displays two bar graphs with information about incoming changes. Incoming changes displays a snapshot of the number of change records currently being read from the source endpoint and written to the target endpoint. To view these graphs, click the **Incoming Changes** bar, shown in the figure below.

Figure 13.11 | Incoming Changes



The following graphs are displayed.

Figure 13.12 | Incoming Change Graphs



The graphs have the following information:

- **Accumulating**: These bars display the number of records currently being read from the source endpoint. These records are accumulated in a queue until they are applied to the target. The following is displayed:
  - **In Memory**: The number of accumulating records that are currently in the computer memory.
  - **On Disk**: The number of accumulating records that are currently stored on disk.
- **Applying**: The number of records currently being written to the target. These are the applied changes. The following is displayed:
  - **In Memory**: The number of records being applied that are currently in the computer

memory.

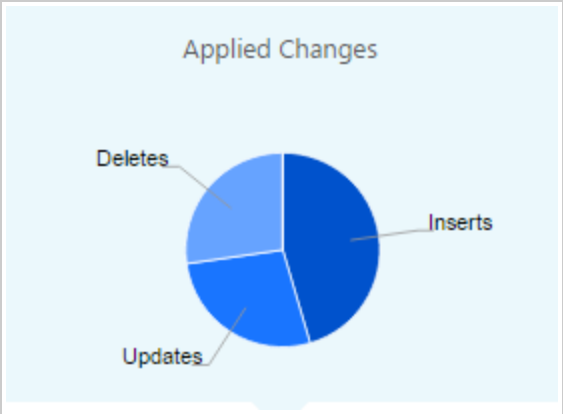
- **On Disk**: The number of records being applied that are currently stored on disk.

Information about Applied Changes

This section displays two tables with information about the applied changes.

To view these tables, click the Applied Changes pie graph, shown in the figure below.

Figure 13.13 | Applied Changes



The following tables are available when you select Applied Changes:

- Recent Activity
- Aggregates

Recent Activity

Click **Recent Activity** at the top of the Applied Changes Details section to view information about which changes occurred in each table. It has the following information:

- **Table Name**: The names of the source tables that are included in the task.
- **Insert**: The number of INSERT operations processed for the specific table.
- **Delete**: The number of DELETE operations processed for the specific table.
- **Update**: The number of UPDATE operations processed for the specific table.
- **DDL**: The number of metadata changes (DDL) processed. DDL changes include information about events like changes to table names or to column names.
- **Total Applied**: The total number of changes applied to the target.
- **Data Errors:** The number of data processing errors for the specific table. Data errors occur at the record level and include conversion errors, errors in transformations, and bad data.

  Resetting the Data Errors Count

  After you have resolved the data errors it is recommended to reset the data errors count. This is especially important if you have configured Replicate to perform an escalation action when the number of errors reaches a certain amount.

Details about the errors can be found in the `attrep_apply_exceptions` control table.

To reset the error count for a specific table, select the table and then click the **Reset data errors** button above the table list. Note that resetting the error count does not delete the error information from the `attrep_apply_exceptions` table.

For information about setting a data error escalation policy, see Data Errors.

For information about the `attrep_apply_exceptions` table, see Apply Exceptions

Note Reloading a table resets the data error count for that table.

- **Last Modified**: The time the last change occurred for the specific table.
- **Reload**: To reload selected tables, select the tables you want to reload and then click the **Reload** button above the table list. When prompted to confirm the operation, click **OK**. The data in the selected tables will be reloaded to the target endpoint. Note that this option is not available for **Apply Changes Only** tasks.

Aggregates

Click Aggregates at the top of the Applied Changes Details section to view information about total changes for each change type and transaction type.

The Aggregate table displays the total changes (for all tables) applied for each of the following types of operations:

- INSERT
- UPDATE
- DELETE
- DDL

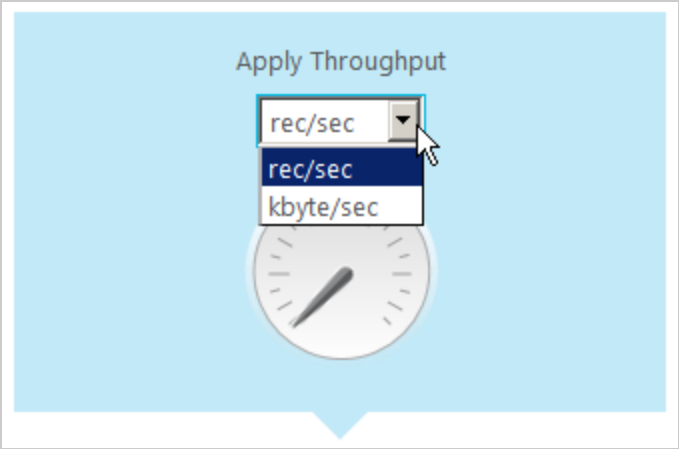
The Aggregate table also displays the information about transactions. It displays the total number and volume of:

- COMMITS
- ROLLBACKS

Information about Change Processing Throughput

Throughput values for apply throughput in a change-processing operation provide information on how fast the change records are loaded to the target endpoint. The information is displayed in a gauge in the Change-Processing graph section. The following figure shows the Apply Throughput gauge:

Figure 13.14 | Apply Throughput Gauge



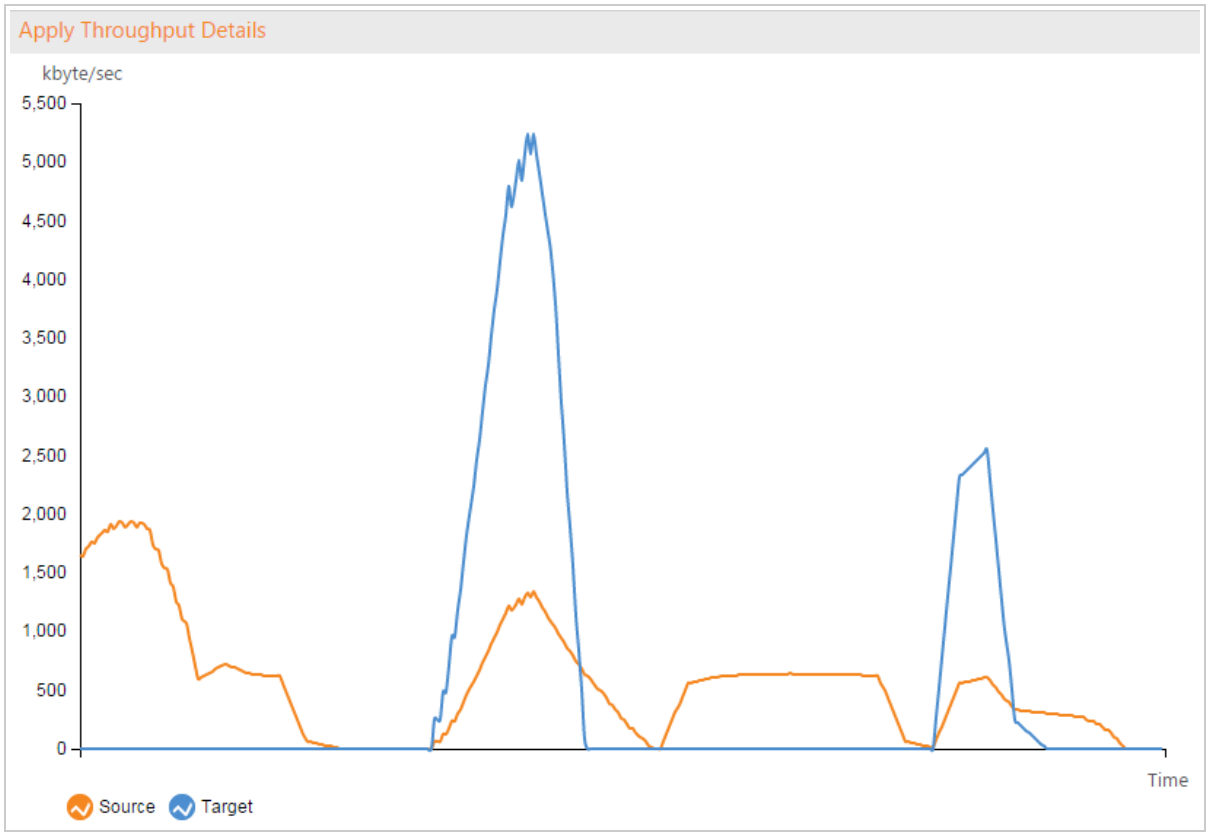
You can set the Apply Throughput measurement values either to the number of change records replicated per second, or to the number of kilobytes replicated per second. The display is always based on the current load operation.

To set the unit of throughput measurement:

- Select either **rec/sec** or **kbyte/sec** from the drop-down menu below the **Apply Throughput** gauge.

Click the **Apply Throughput** gauge to display a graph with the throughput details as shown in the figure below. To view the graph only, click the expand/collapse arrow in right side of the gray bar above the graph. Click the arrow again to restore the progress bars and Change Processing gauges.

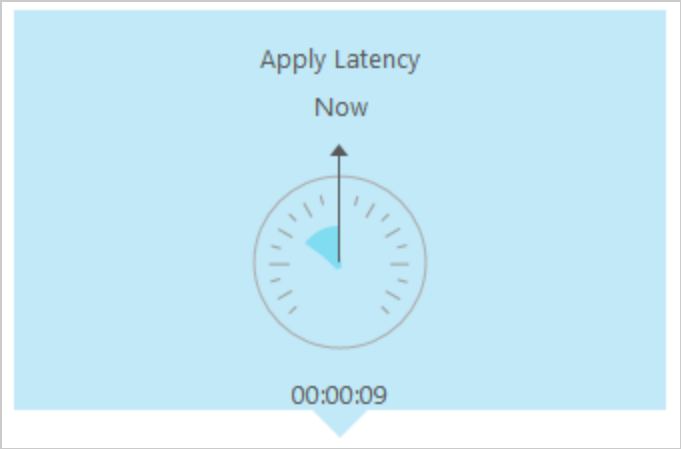
Figure 13.15 | Apply Throughput Details Graph



Information about Apply Latency

Latency values for apply latency in a change-processing operation provide information about the time delay (latency) between the time when a change is visible to the source (and committed), and the time when this same change is visible to the target. The information is displayed in a gauge in the Change-Processing graph section. The following figure shows the Apply Latency gauge.

Figure 13.16 | Apply Latency

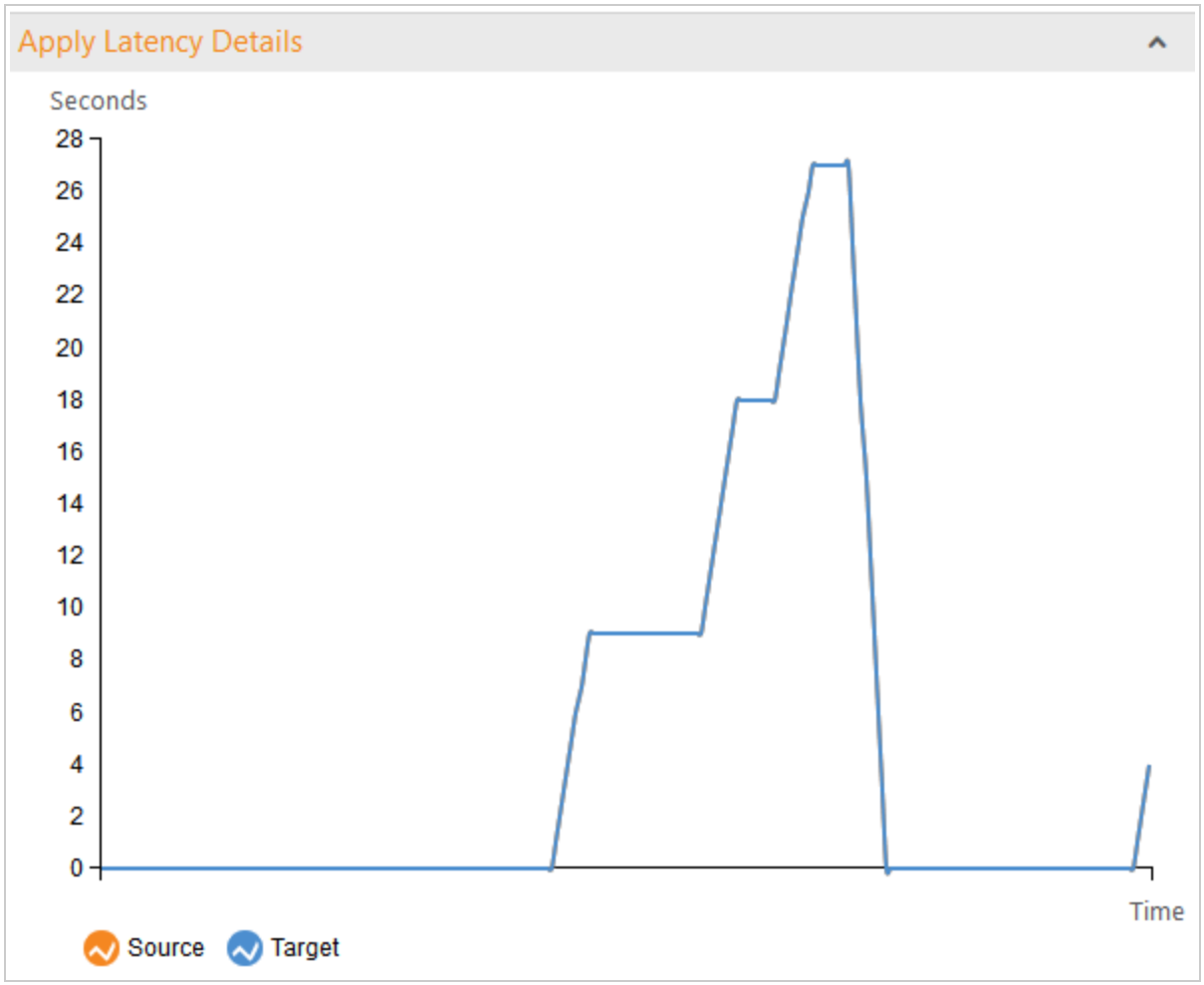


The latency values displayed in the Attunity Replicate Console measure the time delay (latency) between the time when a change is visible to the source (and committed), and the time when this same change is visible to the target. The display is always based on the current change being applied. For more information about latency, see Apply Latency gauge.

Select the Apply Latency gauge to display a graph with the latency details. To view the graph only, click the expand/collapse arrow in right side of the gray bar above the graph. Click the arrow again to restore the progress bars and Change Processing gauges.

Note During data capture, the target latency will always be equal to the source latency (even though no data has reached the target yet). This is simply because target latency is the sum of source latency + apply latency, and can therefore never be less than the source latency.

Figure 13.17 | Apply Latency Details Graph



Viewing Messages

You can see messages sent for the task while in the monitor view. For information on viewing messages, see Reading Messages about a Task.

Using the Monitor Tools

The monitor tools let you view additional information about the task. The following topics describe the information available through these tools:

- Viewing History Information
- Setting the Task Logging Level
- Viewing the Task Log Files and Manually Rolling them Over
- Deleting Log Files

Viewing History Information

The **History** window displays information for each event carried out in the task. To access the **History** information, from Monitor Mode, click **Tools** in the toolbar and then select **History**.

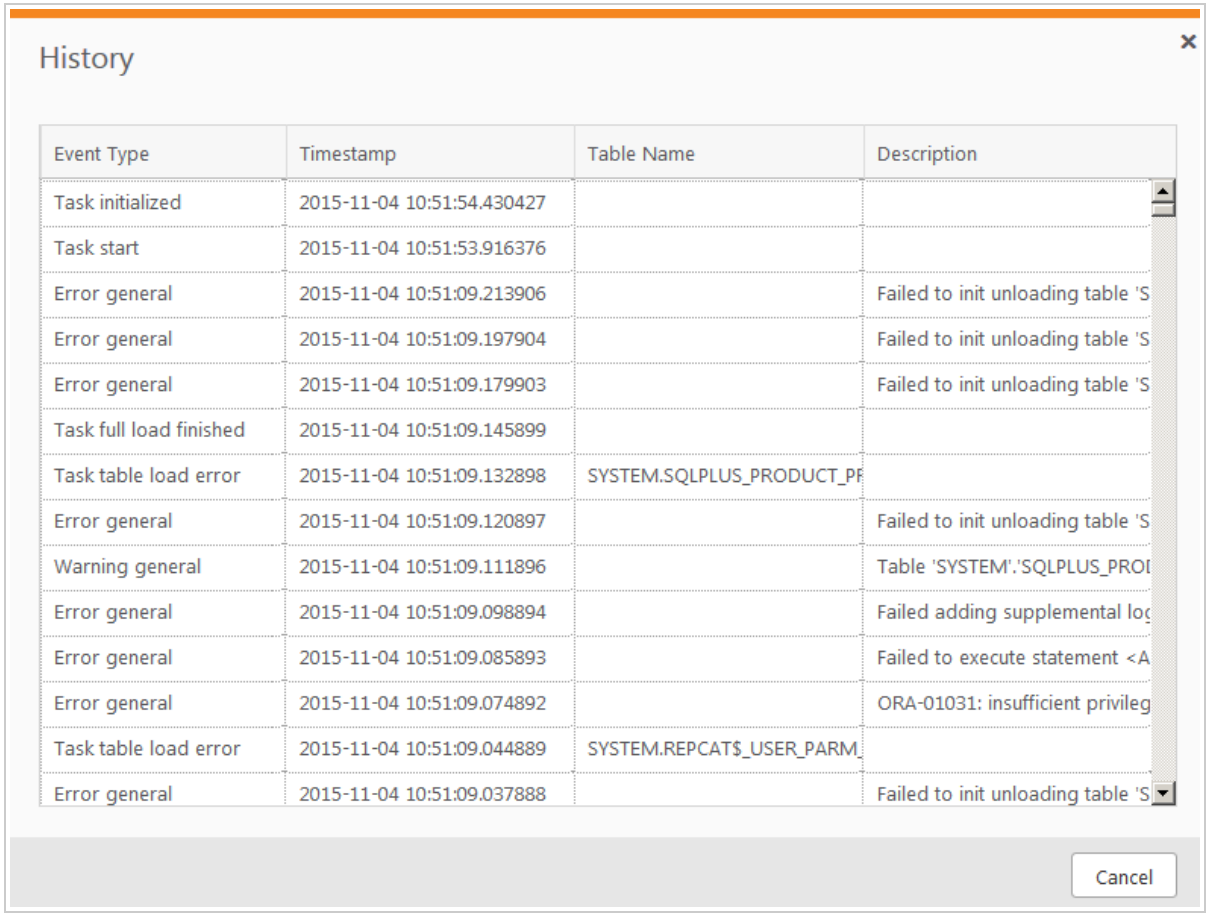
You can view the following information in the **History** window:

- **Event type**: The type of event that occurred, for example **Task started** or T**ask table load finished**.
- **Timestamp**: A timestamp that indicates when the event took place. The timestamp is in the format, `YYYY-MM-DD hh:mm:ss.milliseconds` (to six places).
- **Table Name**: The name of the table where the event takes place if the event is related to a table.
- **Description**: A description of the event. This is not displayed for all events.

  You can double-click the description cell to view a window with the full message if the entire description is not available.

The following figure shows the History window.

Figure 13.18 | History Window



Setting the Task Logging Level

In the **Log Management** window, you can set the logging level for the task you are currently monitoring.

Note The logging level can also be set in the **Logging Level** sub-tab in the **Task Settings** dialog box. For more information, see Logging.

To open the Log Management window:

1. Open the task you are working with if it is not displayed in the Attunity Replicate Console. For information on opening a task, see Editing a Replication Task.
2. Switch to **Monitor** view.
3. Click **Tools** toolbar button and then select **Log Management**.

   The **Log Management** window opens.
4. At the top of the **Log Management** window, set the **Component Logging Level** slider to the log level you want. This sets the logging level for all log modules. Note that all of the sliders for the individual modules move to the same position that you set in the main slider.
5. Make any changes to the sliders for the individual modules. This is optional. Note that if you change the main slider, all of the individual sliders are reset to the new position. If you want to maintain a different logging level for a specific module, you need to reset it.

Viewing the Task Log Files and Manually Rolling them Over

In the **Log Viewer** window, you can view the logs for the task you are currently monitoring and manually roll them over if necessary.

Viewing and Downloading the Task Log Files

Follow the steps below to view or download the task log files.

To open the Log Viewer window:

1. Open the task whose log files you want to view or download.

   For information on opening a task, see Editing a Replication Task.
2. Switch to **Monitor** view.
3. Either, click the **Tools** toolbar button and then select **View Logs**.

   -OR-

   Click the **View Logs** button in the **Messages** pane in the lower right of the console.

   The **Log Viewer** window opens.
4. Select the log file you want to view or download from the list in the **Log Files** pane. If you want to download the file, skip to Step 8.
5. The contents of the log file will be displayed in the right pane. When you select a row in the log file, a tooltip will be display the full message of the selected row.

6. You can browse through the log file using the scroll bar on the right and the navigation buttons at the top of the window.
7. To search for a specific string in the log file, enter the search string in the search box at the top of the window.

   Any terms that match the specified string will be highlighted blue.
8. To download the log file, click the  toolbar button.

   Depending on your browser settings, one of the following will occur:
   - The task JSON file will be automatically downloaded to the default download location
   - You will be prompted for a download location. In this case, save the JSON file to your preferred location.

Manually Rolling Over Task Log Files

You can manually roll the log file for the task you are monitoring in the Log Viewer. This lets you stop logging to the current log file and begin to log to a new log file. The current log file is called `reptask_<name of task>` and saved (older) log files have the file name `reptask_<name of task>_xxxxxxxxxxxx` where `xxxxxxxxxxxx` represents a 12-digit timestamp.

To immediately roll over the selected log file, click the **Roll Log File** button in the top right of the window.

Deleting Log Files

In the **Delete Logs** window, you can manually delete log files older than the specified number of days.

To delete the logs:

1. Open the task whose log files you want to delete.

   For information on opening a task, see Editing a Replication Task.
2. Switch to **Monitor** view.
3. Click the **Tools** toolbar button and then select **Delete Logs**.

   The **Delete Logs** window opens.
4. Optionally change the default number of days (45) and then click the **Delete** button.

   All log files older than the specified number of days will be deleted.

14 | Attunity Replicate Server Settings

This section describes how to configure Attunity Replicate using the Server page. For information on opening and viewing the Server page, see Server View.

Note Configurations made in the Server page affect all Tasks that are created in the Attunity Replicate instance you are working with.

You can do the following in the Server page:

- Create notifications, define mail settings and default recipients for sending notifications
- Register or request a License to work with Attunity Replicate
- Set the error-handling policy for the Attunity Replicate Server
- Set the logging level for the Attunity Replicate Server
- Add File Transfer Service hosts
- Schedule jobs
- Set user permissions
- Set thresholds for disk and memory utilization

In this chapter:

Notifications Settings

The following can be defined in the Notifications settings:

- Defining Notifications
- Setting up Mail Parameters
- Creating a Default Recipient List

To view and edit the Notification settings:

- In **Server** view, click the **Notifications** tab on the left. Then click the **Notifications** sub-tabs to enter your settings.

Defining Notifications

To configure and create notifications, click the **Notifications** sub-tab.

You use notifications to send messages about events that occur when running tasks in Attunity Replicate. Notifications are sent to inform users of any change in the system state, including:

- A task is started or stopped
- Latency is too high
- Memory utilization is too high
- Disk utilization is too high
- An error or a specific type of error occurred

You can manage notifications that you create from the Notifications list. This list provides you with information about each notification defined and lets you activate/deactivate a notification. In addition, you can make changes to the definitions of existing notifications or delete them.

The following topics describe how to define notifications in Attunity Replicate:

- Creating a New Notification
- Using the Notification List
- Editing a Notification
- Deleting a Notification

To open the Notifications page:

From the Server view, click **Notifications** from the menu list at the left. The **Notifications** sub-tab is displayed.

Notifications are sent by:

- An email message to the default list of users and/or to a custom list of users.
- Writing an entry in the Windows Event Log.
- Displaying a message in the Attunity Replicate Console.

Creating a New Notification

Use the **New Notification** wizard to determine the notifications that are sent and who receives them.

To start the New Notification Wizard:

1. In Server view, click the **Notifications** tab on the left and then click the **New Notification** toolbar button.

2. From the drop-down menu, select **Task Events** or **Server Events** according to the notification you want to define.
3. The **New Notification** wizard opens displaying either the **Task Events** or **Server Events** screen (according to your earlier selection).
4. Continue from Creating a Notification for a Task Event or Define the Recipients as appropriate.
5. In the **Notification Name** field, type a name for this notification.
6. Perform the following steps to define the notification:
   - Define the Action that Triggers the Notification
   - Define Which Changes of Status Trigger the Notification
   - Define Errors That Trigger the Notification
   - Define the Notification Distribution Properties
   - Determine the Email-Message Recipients for the Notification
   - Define the Notification Message
   - Associate Tasks with the Notification
   - Review the Notification Rule

Define the Action that Triggers the Notification

In the **Operator** section of the **Task Events** page, you can determine the action that triggers the notification. If the Operator section is not displayed, click on the header with the word Operator to display the options for this section. Select one of the following:

- **Task was started:** To send the notification when the task starts.
- **Task was stopped manually or scheduled:** To send the notification when the task is stopped either manually or by the Scheduler.
- **Task was stopped after Full Load: Cached changes were not applied:** To send the notification when the task is stopped after Full Load completes but before cached changes (changes to the source tables that occurred during Full Load) are applied to the target.
- **Task was stopped after Full Load: Cached changes were applied:** To send the notification when the task is stopped after Full Load completes and cached changes (changes to the source tables that occurred during Full Load) have been applied to the target.
- **Full load started:** To send the notification when the Full Load process starts.
- **Full load completed:** To send the notification when the Full Load process completes.

Once you determine when to send the notification, you can decide whether specific changes in status trigger the notification.

If you want to send a message about problems in latency, memory utilization, or disk utilization, click **Performance/Resources**. See Define Which Changes of Status Trigger the Notification for an explanation.

If you want to send the notification when certain errors occur, click **Errors**. See Define Errors That Trigger the Notification for an explanation.

Or you can click **Next** to Define the Notification Distribution Properties.

Define Which Changes of Status Trigger the Notification

In the **Performance/Resources** section of the **Task Events** page, you can define specific parameters for latency, disk utilization, or memory utilization that trigger a notification.

To set up notifications about latency, disk utilization, or memory utilization:

1. In the New Notification Wizard, **Task Events** page, click **Performance/Resources**.
2. Select one of the following:
   - **Latency is higher than Value seconds**.
   - **Memory utilization exceeded Value MB**
   - **Disk utilization exceeded Value MB**
3. Define the value for the option you select. See the table below for an explanation on each of these options and how to set the value.

Note If you select one of these options, the notification is sent only when the selected parameter is true. However, you must also Define the Action that Triggers the Notification.

Table 14.1 | Set Values for Latency, Disk Utilization, Memory Utilizations

Notification	Set Value	Notes
Latency is higher than Value seconds	Click **[N]** and enter a value in the field that is displayed. Latency is the time interval in seconds between the time a change was committed in the source system and the time it is applied and committed in the target system.	
Clear notification when latency drops below <*n*> seconds.	Use this to set the value that determines when latency returns to "normal limits." Click **[N]** and enter a value.	When latency is below the value entered in this field, it is considered to be in the "normal" range and the notification status ends. If selected, a notification is sent to indicate that latency returned to "normal" status. For more information, see Define

Table 14.1 | Set Values for Latency, Disk Utilization, Memory Utilizations (Cont.)

Notification	Set Value	Notes
		the Notification Message.
Memory utilization exceeded Value MB	Click **[N]** and enter a value in the field that is displayed. Memory utilization is the amount of memory used by the task.	
Clear notification when memory utilization is below <*n*> MB	Use this to set the value that determines when memory utilization returns to "normal limits." Click **[N]** and enter a value.	When memory utilization is below the value entered in this field, it is considered to be in the "normal" range and the notification status ends. For more information, see Define the Notification Message.
Disk utilization exceeded Value MB	Click **[N]** and enter a value in the field that is displayed. Disk utilization is the amount of disk space used. Set a value that indicates that the current amount of disk space used is problematic to running a replication task.	
Clear notification when disk utilization is below <*n*> MB	Use this to set the value that determines when disk utilization returns to "normal limits." Click **[N]** and enter a value.	When disk utilization is below the value entered in this field, it is considered to be in the "normal" range and the notification status ends. For more information, see Define the Notification Message.

Once you determine the status changes that trigger a notification, you can decide whether specific errors trigger a notification.

If you want to send the notification when certain errors occur, click **Errors**. See Define Errors That Trigger the Notification for an explanation.

Or you can click **Next** to Define the Notification Distribution Properties.

Define Errors That Trigger the Notification

In the **Errors** section of the **Task Events** page, you can determine whether notifications are sent when an error occurs. You can determine whether to send the notification for all errors or only for specific error types.

To set up notifications for errors:

1. In the New Notification Wizard, **Task Events** page, click **Errors**.
2. Select one of the following:
   - **Task encountered a non-retriable error and was stopped**: Select this to receive a notification when an error that cannot be retried is returned and a task or tasks are stopped due to this error.
   - **Table encountered more than [N] apply errors**: Select this to receive a notification when a specified number of errors that occur in a CDC operation are applied to a table. In this case, the table is not loaded but the task continues to run.
   - Click **Value** and type the number of errors to trigger the notification. For example, type `50` to send a notification after fifty records fail to be applied to the target table. All apply errors are logged in the `attrep_apply_exceptions` table. The count of apply errors is reset each time the task starts.
   - **Table processing suspended due to errors**: Select this to receive a notification when an error causes a table to stop processing during a full-load operation or suspend CDC. In this case, the table process stops, but the task continues.
   - **Any Error**: Select this to receive a notification when an error occurs in the system.
   - **Any Warning**: Select this to receive a notification when a warning is issued in the system.

Once you determine the error types that trigger a notification, you can:

- Define the Action that Triggers the Notification, if you have not done this already.
- Define Which Changes of Status Trigger the Notification if you have not done this.
- Or you can click **Next** to Define the Notification Distribution Properties.

Define the Notification Distribution Properties

In the **Recipients** page of the New Notification Rule wizard, you determine which users receive the notification and how the notification is sent.

To determine the delivery notification properties:

Select any of the following to determine where (how) the notification is sent:

- **Replication Console**: This selection is selected by default. You cannot change this selection. All notifications are sent to the Replication Console. Notifications are displayed in the **Messages** section for a specific task. This section is displayed in:
  - The Monitor for a specific task. For more information, see Reading Messages about a Task.
  - The right pane of the Tasks page. For more information, see Tasks View.
- **Event Log**: Select this if you want the notification message to be written to the Windows/Linux Event log. For information on how to view the Windows/Linux Event log, see the online help for the version of Windows or Linux you are using.

- **Default Notification Email List**: Select this option if you want to send an email message to the all the recipients on the Default Notification Email List. For more information, see Creating a Default Recipient List.

If you choose to add additional recipients to email notifications or send the email message only to a custom list of recipients, then stay on the **Recipients** page to Determine the Email-Message Recipients for the Notification.

If you do not need to create a custom email recipient list for this notification, click **Next** to Define the Notification Message.

Determine the Email-Message Recipients for the Notification

In addition to sending an email message for a specific notification to the default notification list, you can also create a custom notification list of users who receive this notification only or you can also send the email message to a custom list of users only.

This section describes how:

- To create a custom list of users:
- To send notification email messages to a custom list of recipients only:

To create a custom list of users:

1. In the New Notification Wizard, **Recipients** page, click **Add**. The add button and custom list are in the middle of the **Recipients** page.

   The **Name** field in the first available row in the list is activated.
2. Type the name of the user that you want to receive the message.
3. In the **Name** column of the activated cell, type the name of the user you want to add to the list of default recipients.

   Note If you click another part of the Attunity Replicate Console, the cell will become inactive. You can double-click the cell to enter additional information.
4. Press the [tab] key or double click in the in the **Email** cell, then type the email address for the user you entered in the **Name** cell.
5. Click **Next** to Define the Notification Message.

To send notification email messages to a custom list of recipients only:

1. In the New Notification Wizard, make sure that the **Default Notification Email List** option is *not* selected.
2. Create a Custom Notification list as described in To create a custom list of users:.

Define the Notification Message

You can create a message for your notification. By default, a standard message is created based on the definitions you entered when Define Which Changes of Status Trigger the Notification and Define Errors That Trigger the Notification.

To create a notification message:

1. In the New Notification Wizard, **Message** page, double-click in any of the table cells to open the Edit Notification Message dialog box. See the table Creating a Notification Message for an explanation of the information to enter in each field.
2. Click in the right pane of the dialog box and begin to type your message. In some cases a default message is displayed based on the information you entered in the previous pages of the New Notification Rule wizard. You can edit or delete the message, or create a new message to be sent with the notification in this dialog box.
3. Add variables in messages and email headers you define for notifications, if necessary. You can enter variables in one of two ways:
   - Type a variable into the message pane on the right using the following format:

     {{<VARIABLE_NAME >}}

     For example: `{{TASK_NAME}}`.
   - Use the variables from the left pane of the **Edit Notification** dialog box. To add a variable to the notification message, you can:

     Double-click the variable. The variable is inserted where your cursor is located in the notification message in the right pane.

     Select the variable you want to use and click the arrow key in the middle of the Edit Notification Message dialog box. The variable is inserted where your cursor is located in the notification message in the right pane.

     Drag the variable from the left pane to the location you want to place it in the notification message in the right pane.

     For more information, see the Supported Notification Variables.
4. Click **OK** to enter the message.
5. After you define the message sent with the notification, click **Next** to Associate Tasks with the Notification.

The following table describes how to enter the information in the **Message** page.

Table 14.2 | Creating a Notification Message

To where:	Notification On Message	Notification Off Message
This column describes where the message is sent. For more information, see Define the Notification Distribution Properties.	The **Notification On** Message is sent when the replication task meets the conditions for the notification to be sent. For more information, see Define the Action that Triggers the Notification, Define Which Changes of	The **Notification Off** Message is sent when the replications task returns to its normal state. This type of message is sent for notifications about latency, disk utilization, and memory utilization. For more information, see Define Which Changes of Status Trigger the Notification.

Table 14.2 | Creating a Notification Message (Cont.)

To where:	Notification On Message	Notification Off Message
	Status Trigger the Notification, and Define Errors That Trigger the Notification.	

Console:
The messages in this row are sent to the Attunity Replicate Console. They are displayed in the **Messages** section for a specific task. This section is displayed in:
» The Monitor for a specific task. For more information, see Reading Messages about a Task.
» The right pane of the Tasks page. For more information, see Tasks View.
Note This message is also sent to the Windows Event log if you select this option. For more information, see Define the Notification Distribution Properties.
	In this field, you can edit, change or delete the message that is sent to the Attunity Replicate Console when the replication task meets the conditions for the notification to be sent.
	Example:
	`[{{SERVER_NAME}}\ {{NOTIFICATION_NAME}}] {{TASK_NAME}} replication task latency exceeds defined limits.`
	`Current latency is {{LATENCY}} seconds.`
	This message is sent to the console when latency reaches a value higher than the value you defined.
		In this field, you can edit, change, or delete the message that is sent to the Attunity Replicate Console when the replication task returns to the normal range as you defined when you Define Which Changes of Status Trigger the Notification.
		This field is relevant only for notifications about latency, disk utilization, and memory utilization.
		Example:
		`Latency is back to normal, latency is {{LATENCY}} seconds`
		This message is sent when latency returns to within its normal limits.

Email Subject: This is the subject of the email messages sent for the noti-	In this field, you can edit, change or delete the subject line for an email that is sent when the rep-	In this field, you can edit, change or delete the subject line for an email that is sent when the replication task returns to the normal range as

Table 14.2 | Creating a Notification Message (Cont.)

To where:	Notification On Message	Notification Off Message
fication. See Define the Notification Distribution Properties for information about sending a notification as an email.	lication task meets the conditions for the notification to be sent. **Example**: `[{{SERVER_NAME}}\ {{NOTIFICATION_NAME}}] {{TASK_NAME}} high latency notification` This is the subject for an email message sent when latency reaches a value higher than the value you defined.	you defined when you Define Which Changes of Status Trigger the Notification. This field is relevant only for notifications about latency, disk utilization, and memory utilization. **Example**: `Replicate notification ' {{NOTIFICATION_NAME}}' for task '{{TASK_NAME}}'` This is the subject for an email message sent when latency returns to within its normal limits.

Email Message:
This is the body of the email message sent for the notification.
See Define the Notification Distribution Properties for information about sending a notification as an email.
	In this field, you can edit, change or delete the message that is sent by email when the replication task meets the conditions for the notification to be sent.
	Example:
	`The latency for replication task {{TASK_NAME}} exceeds defined limits.`
	`The current latency is {{LATENCY}} seconds.`
	`----------`
	`----------`
	`----------`
	`This is an automated message generated by Attunity Replicate server {{SERVER_NAME}} for notification {{NOTIFICATION_NAME}}.`
	This is an email message sent when latency reaches a value higher than the
		In this field, you can edit, change, or delete the message that is sent by email when the replication task returns to the normal range as you defined when you Define Which Changes of Status Trigger the Notification.
		This field is relevant only for notifications about latency, disk utilization, and memory utilization.
		Example
		`Latency is back to normal, latency is {{LATENCY}} seconds`
		This is an email message sent when latency returns to within its normal limits.

Table 14.2 | Creating a Notification Message (Cont.)

To where:	Notification On Message	Notification Off Message
	value you defined.	

Event viewer	In this field, you can edit, change or delete the message that is sent to the Windows/Linux event viewer when the replication task meets the conditions for the notification to be sent.
	Note: This field is available only when you select **Event log** when you Define the Notification Distribution Properties.
	Example:
	`[{{SERVER_NAME}}\ {{NOTIFICATION_NAME}}] {{TASK_NAME}} high latency notification`
	`The latency for replication task {{TASK_NAME}} exceeds defined limits.`
	`The current latency is {{LATENCY}} seconds.`
	This message is sent to the event viewer when latency reaches a value higher than the value you defined.	

After you define the message sent with the notification, click **Next** to Associate Tasks with the Notification.

Supported Notification Variables

The table below describes which variables can be included in notification messages.

Table 14.3 | Supported Notification Variables

Variable	Description
LATENCY	The task latency. For a definition of latency, see the Glossary.
MEMORY_USAGE	The amount of memory being consumed by all tasks on the Replicate Server machine.
DISK_USAGE	The amount of disk space being utilized by the task on the Replicate Server machine.
COUNT_ERROR_TABLES	The number of tables in the task with an error status.
ERROR_TABLES	The names of the tables in the task with an error status.
COUNT_ACTIVE_TRANSACTION	The number of open transactions in the task.
COUNT_DATA_ERRORS	The number of data errors encountered by the task. For more information on apply errors, see Global Error Handling
LOADED_RECORDS	The number of records loaded to the target database during the task.
CHANGES_RECORDS	The number of change records processed during the task.
FULLLOAD_COUNT_REQUESTED_TABLES	The number of tables that are queued for loading to the target.
FULLLOAD_COUNT_COMPLETED_TABLES	The number of tables that were loaded to the target.
FULLLOAD_COUNT_ERROR_TABLES	The number of tables that could not be loaded to the target due to error.
FULLLOAD_REQUESTED_TABLES_LIST	The names of the tables that are queued for loading to the target.

Table 14.3 | Supported Notification Variables (Cont.)

Variable	Description
FULLLOAD_COMPLETED_TABLES_LIST	The names of the tables that were loaded to the target.
FULLLOAD_ERROR_TABLES	The names of the tables that could not be loaded to the target due to error.
TABLE_NAME	The name of the table being processed when the notification was sent.
TABLE_OWNER	The owner of the table being processed when the notification was sent.
RECORD_COUNTER	The number of records that had been processed when the notification was sent.
ERROR_TEXT	The error message when a task ends with an error.
TASK_NAME	The name of the task.
NOTIFICATION_NAME	The name of the notification.
TABLE_COUNT_APPLY_ERRORS	The number of tables with apply errors. For more information on apply errors, see Global Error Handling
SERVER_NAME	The host name of the Replicate Server machine.
STORAGE_UTILIZATION_OLD_STATE	The storage utilization state before it moved to the "new" state described below. For a description of possible states, see Define the Event that Triggers the Notification.
STORAGE_UTILIZATION_NEW_STATE	The storage utilization state after it moved from the "old" state described above. For a description of possible states, see Define the Event that

Table 14.3 | Supported Notification Variables (Cont.)

Variable	Description
	Triggers the Notification.
USED_STORAGE_UTILIZATION	The amount of disk space (on the drive where the Replicate Data folder is located) used by all tasks.
TOTAL_STORAGE_UTILIZATION	The total amount of disk space available on the drive where the Replicate Data folder is located.

Associate Tasks with the Notification

By default, notifications are sent for all tasks that are defined in the Attunity Replicate instance you are using. You can determine whether to send the notification to specific tasks defined in the Attunity Replicate instance you are using. For example, you can define a different latency rate for a specific task that is replicating from a slow system.

To associate the notification with tasks:

1. In the New Notification Wizard, **Associate** page, select one of the following:
   - **All Tasks**: To associate this notification with all tasks that are defined in the Attunity Replicate instance you are working with. In this case all tasks that were previously defined and any future task will be associated with this notification.

     If you choose to associate this notification with All Tasks, then click Next to Review the Notification Rule.
   - **Selected Tasks**: To associate this notification with one or more specific tasks only. Continue with the next step.
2. Select the check box next to any of the tasks you want to associate with this notification. You can select one or more tasks.

   Note The Task check box at the top of the check-box column lets you select all of the tasks that are displayed. When you select this check box it is as if you select each of the tasks individually. Therefore, if you add tasks in the future they will not be included.
3. Click **Next** to Review the Notification Rule.

Review the Notification Rule

The **Summary** page lets you review the notification rule that you defined so that you can determine whether the selections you made in the wizard are correct. If you want to make changes, click **Back** and go to the page or pages you want to change.

When you are sure that the notification rule is defined in the way that you want, click **Finish** to close the wizard and add the rule to the notification list (seeUsing the Notification List).

After you close the wizard, make sure to click **Save** at the top of the **Settings** page. This will save the information for *all* settings, not only for the notification rule that you created. If you made changes that you do not want to keep, click **Discard** to discard all changes before you make changes to any of the other settings.

Creating a Notification for a Task Event

Use the **New Notification** wizard to create notifications for task-based events.

To create a notification for a task event

1. Launch the New Notification wizard as described in Creating a New Notification.
2. In the **Notification Name** field, type a name for the notification.
3. Perform the following steps to define the notification:
   - Define the Action that Triggers the Notification
   - Define Which Changes of Status Trigger the Notification
   - Define Errors That Trigger the Notification
   - Define the Recipients
   - Define the Notification Message
   - Associate Tasks with the Notification
   - Review the Notification Rule

Define the Action that Triggers the Notification

In the **Operator** section of the **Task Events** page, you can determine the action that triggers the notification. If the Operator section is not displayed, click on the header with the word Operator to display the options for this section. Select one of the following:

- **Task was started:** To send the notification when the task starts.
- **Task was stopped manually or scheduled:** To send the notification when the task is stopped either manually or by the Scheduler.
- **Task was stopped after Full Load: Cached changes were not applied:** To send the notification when the task is stopped after Full Load completes but before cached changes (changes to the source tables that occurred during Full Load) are applied to the target.
- **Task was stopped after Full Load: Cached changes were applied:** To send the notification when the task is stopped after Full Load completes and cached changes (changes to the source tables that occurred during Full Load) have been applied to the target.
- **Full load started:** To send the notification when the Full Load process starts.
- **Full load completed:** To send the notification when the Full Load process completes.

Once you determine when to send the notification, you can decide whether specific changes in status trigger the notification.

If you want to send a message about problems in latency, memory utilization, or disk utilization, click **Performance/Resources**. See Define Which Changes of Status Trigger the Notification for an explanation.

If you want to send the notification when certain errors occur, click **Errors**. See Define Errors That Trigger the Notification for an explanation.

Or you can click **Next** to Define the Recipients.

Define Which Changes of Status Trigger the Notification

In the **Performance/Resources** section of the **Task Events** page, you can define specific parameters for latency, disk utilization, or memory utilization that trigger a notification.

To set up notifications about latency, disk utilization, or memory utilization:

1. In the New Notification Wizard, **Task Events** page, click **Performance/Resources**.
2. Select one of the following:
   - **Latency is higher than Value seconds**.
   - **Memory utilization exceeded Value MB**
   - **Disk utilization exceeded Value MB**
3. Define the value for the option you select. See the table below for an explanation on each of these options and how to set the value.

Note If you select one of these options, the notification is sent only when the selected parameter is true. However, you must also Define the Action that Triggers the Notification.

Table 14.4 | Set Values for Latency, Disk Utilization, Memory Utilization

Notification	Set Value	Notes
Latency is higher than Value seconds	Click **[N]** and enter a value in the field that is displayed. Latency is the time interval in seconds between the time a change was committed in the source system and the time it is applied and committed in the target system.	
Clear notification when latency drops	Use this to set the value that determines when latency returns to "normal limits."	When latency is below the value entered in this field, it is considered to be in the "normal"

Table 14.4 | Set Values for Latency, Disk Utilization, Memory Utilization (Cont.)

Notification	Set Value	Notes
below <*n*> seconds.	Click **[N]** and enter a value.	range and the notification status ends. If selected, a notification is sent to indicate that latency returned to "normal" status. For more information, see Define the Notification Message.
Memory utilization exceeded Value MB	Click **[N]** and enter a value in the field that is displayed. Memory utilization is the amount of memory used by the task.	
Clear notification when memory utilization is below <*n*> MB	Use this to set the value that determines when memory utilization returns to "normal limits." Click **[N]** and enter a value.	When memory utilization is below the value entered in this field, it is considered to be in the "normal" range and the notification status ends. For more information, see Define the Notification Message.
Disk utilization exceeded Value MB	Click **[N]** and enter a value in the field that is displayed. Disk utilization is the amount of disk space used. Set a value that indicates that the current amount of disk space used is problematic to running a replication task.	
Clear notification when disk utilization is below <*n*> MB	Use this to set the value that determines when disk utilization returns to "normal limits." Click **[N]** and enter a value.	When disk utilization is below the value entered in this field, it is considered to be in the "normal" range and the notification status ends. For more information, see Define the Notification Message.

Once you determine the status changes that trigger a notification, you can decide whether specific errors trigger a notification.

If you want to send the notification when certain errors occur, click **Errors**. See Define Errors That Trigger the Notification for an explanation.

Or you can click **Next** to Define the Recipients.

Define Errors That Trigger the Notification

In the **Errors** section of the **Task Events** page, you can determine whether notifications are sent when an error occurs. You can determine whether to send the notification for all errors or only for specific error types.

To set up notifications for errors:

1. In the New Notification Wizard, **Task Events** page, click **Errors**.
2. Select one of the following:
   - **Task encountered a non-retriable error and was stopped**: Select this to receive a notification when an error that cannot be retried is returned and a task or tasks are stopped due to this error.
   - **Table encountered more than [N] apply errors**: Select this to receive a notification when a specified number of errors that occur in a CDC operation are applied to a table. In this case, the table is not loaded but the task continues to run.
   - Click **Value** and type the number of errors to trigger the notification. For example, type `50` to send a notification after fifty records fail to be applied to the target table. All apply errors are logged in the `attrep_apply_exceptions` table. The count of apply errors is reset each time the task starts.
   - **Table processing suspended due to errors**: Select this to receive a notification when an error causes a table to stop processing during a full-load operation or suspend CDC. In this case, the table process stops, but the task continues.
   - **Any Error**: Select this to receive a notification when an error occurs in the system.
   - **Any Warning**: Select this to receive a notification when a warning is issued in the system.

Once you determine the error types that trigger a notification, you can:

- Define the Action that Triggers the Notification, if you have not done this already.
- Define Which Changes of Status Trigger the Notification if you have not done this.
- Or you can click **Next** to Define the Recipients.

Define the Recipients

Notifications are always displayed in **Notifications** tab in the **Messages** panel of the Replicate Console. In the **Recipients** page of the New Notification wizard, you can determine whether to also send the notification to Windows Event Log and/or email recipients.

To determine the notification recipients:

Select any of the following to determine the notification recipients:

- **Event log**: Select this if you want the notification message to be written to the Windows/Linux Event log. For information on how to view the Windows/Linux Event log, see the online help for the version of Windows or Linux you are using.

- **Default notification email list**: Select this option if you want to send an email message to the all the recipients on the Default Notification Email List. For more information, see Creating a Default Recipient List. To see the current list of default recipients, click **Show List.**
- **Custom email recipients:** Select this option to send the email notification to specific recipients.

  Then:

  - Click the **Add Recipient** button.

    The **Name** field in the first available row in the list is activated.
  - Type the name of the user that you want to receive the message.

    Note If you click another part of the Attunity Replicate Console, the cell will become inactive. You can double-click the cell to enter additional information.
  - Press the [tab] key or double click in the in the **Email** cell, then type the email address for the user you entered in the **Name** cell.
  - Repeat the above to add more recipients.

When you have finished, click **Next** to Define the Notification Message.

Define the Notification Message

You can create a custom message for your notification. By default, a standard message is created based on your settings in the Task Events screen.

To create a notification message:

1. In the New Notification Wizard, **Message** page, double-click in any of the table cells to open the Edit Notification Message dialog box. See the table Creating a Notification Message for an explanation of the information to enter in each field.
2. Click in the right pane of the dialog box and begin to type your message. In some cases a default message is displayed based on the information you entered in the previous pages of the New Notification Rule wizard. You can edit or delete the message, or create a new message to be sent with the notification in this dialog box.
3. Add variables in messages and email headers you define for notifications, if necessary. You can enter variables in one of two ways:
   - Type a variable into the message pane on the right using the following format:

     {{<VARIABLE_NAME >}}

     For example: `{{TASK_NAME}}`.
   - Use the variables from the left pane of the **Edit Notification** dialog box. To add a variable to the notification message, you can:

     Double-click the variable. The variable is inserted where your cursor is located in the notification message in the right pane.

Select the variable you want to use and click the arrow key in the middle of the **Edit Notification Message** dialog box. The variable is inserted where your cursor is located in the notification message in the right pane.

Drag the variable from the left pane to the location you want to place it in the notification message in the right pane.

For more information, see the Supported Notification Variables.

4. Click **OK** to enter the message.
5. After you define the message sent with the notification, click **Next** to Associate Tasks with the Notification.

The following table describes how to enter the information in the **Message** page.

Table 14.5 | Creating a Notification Message

To where:	Notification On Message	Notification Off Message
This column describes where the message is sent. For more information, see Define the Recipients.	The **Notification On** Message is sent when the replication task meets the conditions for the notification to be sent. For more information, see Define the Action that Triggers the Notification, Define Which Changes of Status Trigger the Notification, and Define Errors That Trigger the Notification.	The **Notification Off** Message is sent when the replication task returns to its normal state. This type of message is sent for notifications about latency, disk utilization, and memory utilization. For more information, see Define Which Changes of Status Trigger the Notification.
Console: The messages in this row are sent to the Attunity Replicate Console. They are displayed in the **Messages** section for a specific task. This section is displayed in: ▪ The Monitor for a specific task. For more information, see Reading Messages about a Task.	In this field, you can edit, change or delete the message that is sent to the Attunity Replicate Console when the replication task meets the conditions for the notification to be sent. **Example**: `[{{SERVER_NAME}}\ {{NOTIFICATION_NAME}}] {{TASK_NAME}} replication task latency exceeds defined limits.`	In this field, you can edit, change, or delete the message that is sent to the Attunity Replicate Console when the replication task returns to the normal range as you defined when you Define Which Changes of Status Trigger the Notification. This field is relevant only for notifications about latency, disk utilization, and memory utilization. **Example**: `Latency is back to normal, latency is {{LATENCY}} seconds`

Table 14.5 | Creating a Notification Message (Cont.)

To where:	Notification On Message	Notification Off Message
▪ The right pane of the Tasks page. For more information, see Tasks View. **Note** This message is also sent to the Windows Event log if you select this option. For more information, see Define the Recipients.	`Current latency is {{LATENCY}} seconds.` This message is sent to the console when latency reaches a value higher than the value you defined.	This message is sent when latency returns to within its normal limits.
Email Subject: This is the subject of the email messages sent for the notification. See Define the Recipients for information about sending a notification as an email.	In this field, you can edit, change or delete the subject line for an email that is sent when the replication task meets the conditions for the notification to be sent. **Example**: `[{{SERVER_NAME}}\ {{NOTIFICATION_NAME}}] {{TASK_NAME}} high latency notification` This is the subject for an email message sent when latency reaches a value higher than the value you defined.	In this field, you can edit, change or delete the subject line for an email that is sent when the replication task returns to the normal range as you defined when you Define Which Changes of Status Trigger the Notification. This field is relevant only for notifications about latency, disk utilization, and memory utilization. **Example**: `Replicate notification '{{NOTIFICATION_NAME}}' for task '{{TASK_NAME}}'` This is the subject for an email message sent when latency returns to within its normal limits.
Email Message: This is the body of the email message sent for the notification. See Define the Recipients for information about sending a noti-	In this field, you can edit, change or delete the message that is sent by email when the replication task meets the conditions for the notification to be sent. **Example**:	In this field, you can edit, change, or delete the message that is sent by email when the replication task returns to the normal range as you defined when you Define Which Changes of Status Trigger the Notification. This field is relevant only for noti-

Table 14.5 | Creating a Notification Message (Cont.)

To where:	Notification On Message	Notification Off Message

fication as an email.	`The latency for replication task {{TASK_NAME}} exceeds defined limits.`
	`The current latency is {{LATENCY}} seconds.`
	`----------`
	`This is an automated message generated by Attunity Replicate server {{SERVER_NAME}} for notification {{NOTIFICATION_NAME}}.`
	This is an email message sent when latency reaches a value higher than the value you defined.
		fications about latency, disk utilization, and memory utilization.
		Example
		`Latency is back to normal, latency is {{LATENCY}} seconds`
		This is an email message sent when latency returns to within its normal limits.

Event viewer	In this field, you can edit, change or delete the message that is sent to the Windows/Linux event viewer when the replication task meets the conditions for the notification to be sent.
	Note: This field is available only when you select **Event log** when you Define the Recipients.
	Example:
	`[{{SERVER_NAME}}\ {{NOTIFICATION_NAME}}] {{TASK_NAME}} high latency notification`
	`The latency for replication task {{TASK_NAME}} exceeds defined limits.`	

Table 14.5 | Creating a Notification Message (Cont.)

To where:	Notification On Message	Notification Off Message
	`The current latency is {{LATENCY}} seconds.` This message is sent to the event viewer when latency reaches a value higher than the value you defined.	

After you define the message sent with the notification, click **Next** to Associate Tasks with the Notification.

Supported Notification Variables

For information on the variables that can be included in a notification, see Supported Notification Variables.

Associate Tasks with the Notification

By default, notifications are sent for all tasks that are defined in the Attunity Replicate instance you are using. You can determine whether to send the notification to specific tasks defined in the Attunity Replicate instance you are using. For example, you can define a different latency rate for a specific task that is replicating from a slow system.

To associate the notification with tasks:

1. In the New Notification Wizard, **Associate** page, select one of the following:
   - **All Tasks**: To associate this notification with all tasks that are defined in the Attunity Replicate instance you are working with. In this case all tasks that were previously defined and any future task will be associated with this notification.

     If you choose to associate this notification with All Tasks, then click Next to Review the Notification Rule.
   - **Selected Tasks**: To associate this notification with one or more specific tasks only. Continue with the next step.
2. Select the check box next to any of the tasks you want to associate with this notification. You can select one or more tasks.

   Note The Task check box at the top of the check-box column lets you select all of the tasks that are displayed. When you select this check box it is as if you select each of the tasks individually. Therefore, if you add tasks in the future they will not be included.
3. Click **Next** to Review the Notification Rule.

Review the Notification Rule

The **Summary** page lets you review the notification rule that you defined so that you can determine whether the selections you made in the wizard are correct. If you want to make changes, click **Back** and go to the page or pages you want to change.

When you are sure that the notification rule is defined in the way that you want, click **Finish** to close the wizard and add the rule to the notification list (seeUsing the Notification List).

After you close the wizard, make sure to click **Save** at the top of the **Settings** page. This will save the information for *all* settings, not only for the notification rule that you created. If you made changes that you do not want to keep, click **Discard** to discard all changes before you make changes to any of the other settings.

Creating a Notification for a Server Event

Use the **New Notification** wizard to create notifications for server-based events.

To create a notification for a server event

1. Launch the New Notification wizard as described in Creating a New Notification.
2. In the **Notification Name** field, type a name for the notification.
3. Perform the following steps to define the notification:
   a. Define the Event that Triggers the Notification
   b. Define the Recipients
   c. Define the Notification Message
   d. Review the Notification Rule

Define the Event that Triggers the Notification

In the **Disk Space** section, you can determine the disk space utilization event that triggers the notification.

Select one of the following:

- **Disk space utilization reaches the high threshold:** The notification will be triggered when disk space utilization reaches the percentage defined for the high threshold.
- **Disk space utilization reaches the critical threshold:** The notification will be triggered when disk space utilization reaches the percentage defined for the critical threshold.
- **Disk space utilization returns to normal:** The notification will be triggered when disk space utilization returns to normal percentage (i.e. not high or critical).
- **Disk space utilization reaches any of the defined thresholds or returns to normal:** The notification will be triggered in any of the following scenarios:
  - Disk space utilization increases from normal to the high threshold
  - Disk space utilization increases from normal to the critical threshold

- Disk space utilization increases from the high threshold to the critical threshold
- Disk space utilization returns to normal from the high threshold
- Disk space utilization returns to normal from the critical threshold
- Disk space utilization returns to the high threshold from the critical threshold

In the **System Memory** section, you can determine the system memory utilization event that triggers the notification.

Select one of the following:

- **System memory utilization reaches the high threshold:** The notification will be triggered when system memory utilization reaches the percentage defined for the high threshold.
- **System memory utilization reaches the critical threshold:** The notification will be triggered when system memory utilization reaches the percentage defined for the critical threshold.
- **System memory utilization returns to normal:** The notification will be triggered when system memory utilization returns to normal percentage (i.e. not high or critical).
- **System memory utilization reaches any of the defined thresholds or returns to normal:** The notification will be triggered in any of the following scenarios:
  - System memory utilization increases from normal to the high threshold
  - System memory utilization increases from normal to the critical threshold
  - System memory utilization increases from the high threshold to the critical threshold
  - System memory utilization returns to normal from the high threshold
  - System memory utilization returns to normal from the critical threshold
  - System memory utilization returns to the high threshold from the critical threshold

Click **Next** to Define the Recipients.

Define the Recipients

For more information, see Define the Recipients.

Define the Notification Message

You can create a custom message for your notification. By default, a standard message is created based on your settings in the **Server Events** screen.

To edit the default notification message:

1. In the New Notification Wizard, **Message** page, click the message text to open the **Edit Notification Message** dialog box. See the table Creating a Notification Message for an explanation of the information to enter in each field.
2. Click in the right pane of the dialog box and begin to type your message. In some cases a default message is displayed based on the information you entered in the previous pages of the New Notification Rule wizard. You can edit the message or create a new message to be sent with the notification in this dialog box.
3. Optionally, add variables in messages and email headers you define for notifications.

You can enter variables in one of two ways:

- Type a variable into the message pane on the right using the following format:
  `{{<VARIABLE_NAME >}}`
  For example: `{{TASK_NAME}}`.
- Use the variables from the left pane of the **Edit Notification Message** dialog box. To add a variable to the notification message, you can:
  Double-click the variable. The variable is inserted where your cursor is located in the notification message in the right pane.
  Select the variable you want to use and click the arrow key in the middle of the **Edit Notification Message** dialog box. The variable is inserted where your cursor is located in the notification message in the right pane.
  Drag the variable from the left pane to the location you want to place it in the notification message in the right pane.
  For more information, see the Supported Notification Variables.

4. Click **OK** to enter the message.
5. After you define the message sent with the notification, click **Next** to Review the Notification Rule.

The following table describes how to enter the information in the **Message** page.

Table 14.6 | Creating a Notification Message

To where:	Notification Message
This column describes where the message is sent. For more information, see Define the Recipients.	Sent when the server meets the conditions for the notification to be sent.
Console: The messages in this row are sent to the Console. **Note** The same message is also sent to the Windows Event Log if you chose to send messages to the Windows Event Log. For more information, see Define the Recipients.	In this field, you can edit the message that is sent to the Attunity Replicate Console when the server meets the conditions for the notification to be sent.
Email Subject: This is the subject of the email messages sent for the notification. See Define the Recipients for information	In this field, you can edit the subject line for an email that is sent when the server meets the conditions for the notification to be sent.

Table 14.6 | Creating a Notification Message (Cont.)

To where:	Notification Message
about sending a notification as an email.	
Email Message: This is the body of the email message sent for the notification. See Define the Recipients for information about sending a notification as an email.	In this field, you can edit the message that is sent by email when the server meets the conditions for the notification to be sent.

Supported Notification Variables

For information on the variables that can be included in a notification, see Supported Notification Variables.

Review the Notification Rule

The **Summary** page lets you review the notification rule that you defined so that you can determine whether the selections you made in the wizard are correct. If you want to make changes, click **Back** and go to the page or pages you want to change.

When you are sure that the notification rule is defined in the way that you want, click **Finish** to close the wizard and add the rule to the notification list (seeUsing the Notification List).

After you close the wizard, make sure to click **Save** at the top of the **Settings** page. This will save the information for *all* settings, not only for the notification rule that you created. If you made changes that you do not want to keep, click **Discard** to discard all changes before you make changes to any of the other settings.

Using the Notification List

The Notification List lists all of the notification rules that are defined for the Attunity Replicate instance you are working with. It has the following information:

- **Name**: Displays the name of the notification rule.
- **Condition**: Displays the condition that triggers the notification to be sent. For more information, see Define Which Changes of Status Trigger the Notification.
- **Send Message To**: Displays custom users that receive the message. For more information, see Determine the Email-Message Recipients for the Notification.
- **Tasks**: Displays the tasks that are associated with this notification rule. For more information, see Associate Tasks with the Notification.
- **Active**: Select the check box in this column to activate the notification. If this check box is cleared, notifications defined by this rule are not sent. This check box is selected by default.

Editing a Notification

You can make changes to any notification rule.

To edit a notification rule:

1. From the Notification List, select the notification you want to edit.
2. Click **Open** (at the top of the list).

   or

   Double-click the notification you want to edit.

   The Edit Notification Rule wizard opens.
3. Make any changes you need in the wizard. For information on how to work with each of the pages in the New Notification Rule wizard, see Creating a New Notification.

Note You can only make changes to those sections that you defined when Creating a New Notification.

- You cannot change name of the notification.
- If you defined a notification to let you know when the task or full load started or stopped, this cannot be edited. For example, if you created a notification rule for starting a task and you now also want to get notified when the task stops, you must create a new notification rule.
- In the Notify When? page, you can make changes to the data you defined in the original notification rule. For example, if you defined a **Memory utilization** message in the **Notify when?** page, **Performance/Resources** section, you can only change this parameter. If you want to add information about something that was not defined in the original notification rule, for example you want to add errors to your notification or you want to get information about latency, you must create a new notification rule.

Deleting a Notification

You can delete notification rules that you no longer want to use.

Note When you delete a notification, it is deleted permanently.

To delete a notification:

1. From the Notification List select the notification you want to delete.
2. Click **Delete** (at the top of the list).

Setting up Mail Parameters

The Mail parameters define the mail server used to send notifications.

To set the Mail parameters:

1. Click the **Mail Settings** sub-tab and enter the following information:
   - **Mail server**: Type the outgoing mail server you are using to send the notifications that you define in Attunity Replicate, for example, `smtp.example.com`.
   - **Port**: Type the port number where the mail server is located. The default value is **25**.
   - **Use SSL**: Select this check box if you want to use SSL security to connect to the mail server for the notifications that are sent.
   - **Anonymous login**: Check this to allow an Attunity Replicate user to access the mail server to receive messages without having to provide any user credentials.
   - **User name**: Type an email user name for the user account that is sending the notifications. For SMTP authentication be sure to supply a valid user name.
   - **Password**: Type the password for the email user account that is sending the notifications. For SMTP authentication be sure that password provided is valid.
   - **Sender email address**: Enter the email address that sends the email notifications. This is the address that appears in the **From** field of the email notification.
   - **Send test email**: Click to open the **Send Test Email** dialog box.
   - **Email address for test email**: Type an email address to receive a test email message from the server you configured. Use this to determine that the **Mail Parameters** you defined are valid.
2. Click **Save** at the top of the screen to save *all* of the changes you made, not only for the recipient list.

> **Note** If you made changes that you do not want to keep, click **Discard** to discard all changes before you make changes to any of the other settings.

Creating a Default Recipient List

Click the **Default Recipients List** sub-tab to create a default recipient list.

A default recipient list is a list of recipients that receive all of the notifications that you define for task-based or server-based events. This allows you to use one list for all email notifications without having to define the list each time you create a notification.

> **Note** You can choose to send notifications to a different list or to additional users for any specific notification. You define these exceptions when you create the specific notification. For more information, see Define the Notification Distribution Properties.

To create a Default Recipient List:

1. At the top of the **Default Recipient List** settings page, click **Add Recipient**.
   The next row in the Recipient List table becomes available.
2. Type the name of the user you want to add to the list of default recipients. Continue to enter a name and email address for each recipient you want to include in the default list.

3. Press the [tab] key or double click in the in the **Email** cell, then type the email address for the user you entered in the **Name** cell.
4. Click **Save** at the top of the screen to save all of the changes you made.

> **Note** Click **Save** to save the information for *all* settings, not only for the recipient list. If you made changes that you do not want to keep, click **Discard** to discard all changes before you make changes to any of the other settings.

License Settings

You need to register the software before you can use Attunity Replicate. Your Attunity vendor should provide you with a text file called license.txt. This file contains details such as the product expiration date (if any).

Use the License settings page for:

- Requesting a License
- Registering a License
- Viewing a License

To open the License settings page:

- From the Server view, click **License** from the menu list at the left. The **License** sub-tab is displayed.

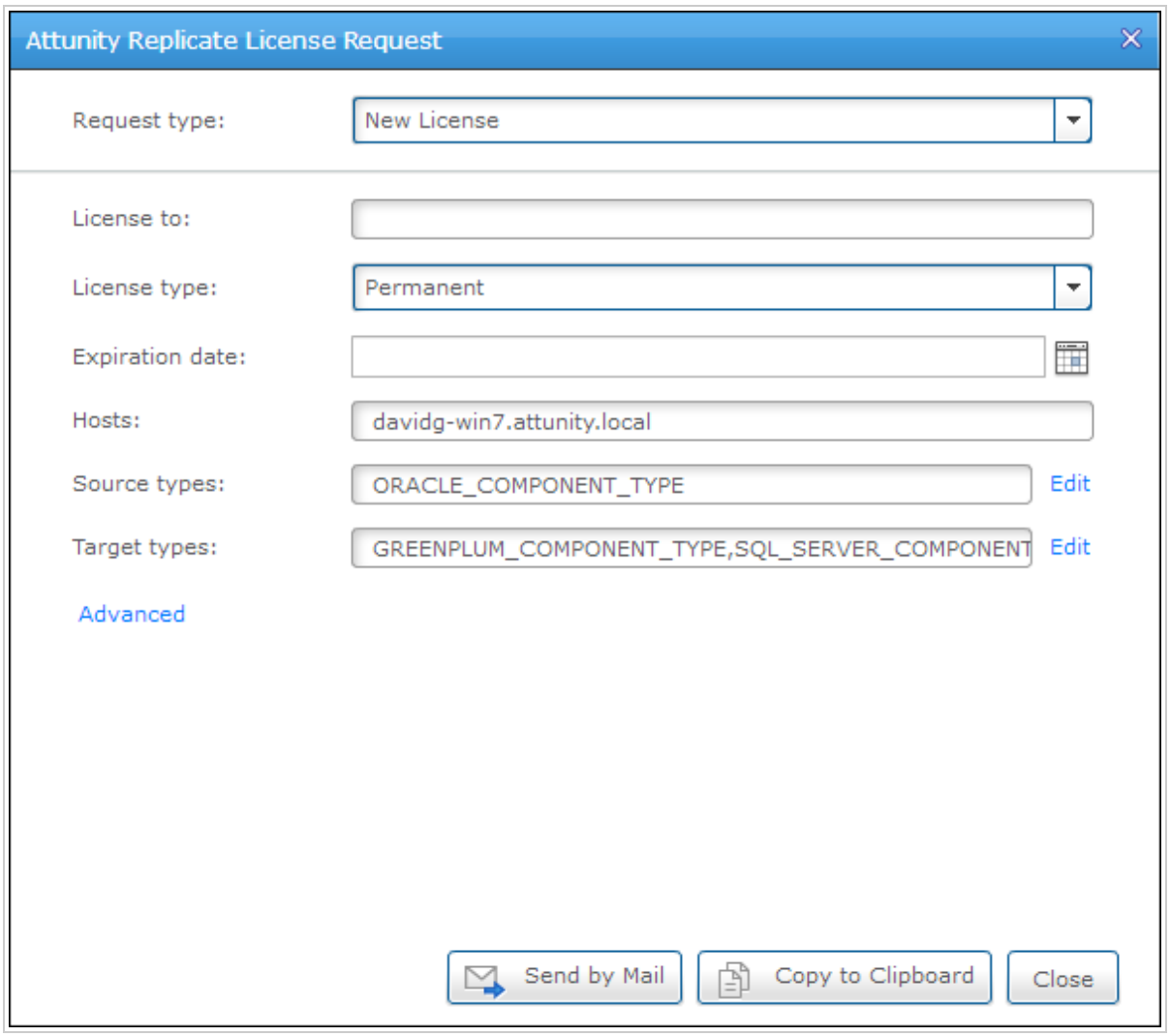
Requesting a License

You must have a valid license to work with Attunity Replicate. You can request a license from the License settings page in the Attunity Replicate Console. In the License Request dialog box, fill out the required information and submit the request by email. Once your request is approved, the license file is sent to you by email. To use Attunity Replicate, register the license by using the procedure described in Registering a License.

To request a license:

1. From the **Server** page, click **License**.
2. At the top of the **License** tab, click **Request License**.

   The **Replication License Request** dialog box opens.



3. Enter the requested information:
   - Request type: Select one of the following:
     - **New License**: Select this if this is your initial license request.
     - **Extend License**: Select this if you have a license and want to extend its period of validity.
     - **Alter License**: Select this if you want to make changes to an existing license. For example, if you want to add additional sources or targets or change the host computer.
   - **License to**: Type the name of the company or group that is requesting a license to use Attunity Replicate.
   - **License type**: Select one of the following:
     - **Permanent**: Select this if the license will always be valid. Permanent licenses do not require an expiration date.

- **Evaluation**: Select this if you are requesting a temporary license to use Attunity Replicate for a trial period.
- **Term**: Select this if you are requesting a license that is valid for a specific period of time. In this case you must be sure to include an expiration date in your request.

- **Expiration date**: Click in this field to select the expiration date using the pop-up calendar. This is required only if you selected **Evaluation** or **Term** in as the License type.
- **Hosts**: Type the name of the local computer where Attunity Replicate is installed. By default the name of the local computer is displayed in this field. You can change this or add additional computers if you are installing Attunity Replicate in a different or an additional location.
- **Source Types**: Click **Edit** to open the Edit Source Types dialog box. Check the endpoint types you are working with as your replication sources. You can select one or more endpoint endpoints as necessary. If you need to work with all available endpoints, click **All**.
- **Target Types**: Click **Edit** to open the Edit Target Types dialog box. Check the endpoint types you are working with as your replication targets. You can select one or more endpoint endpoints as necessary. If you need to work with all available sources, click **All**.

4. Click **Send by Mail** to open an email request for the license. Send the email to the address entered in the recipient field of your default email client.

   Note If you have not registered a default email client, clicking the **Send by Mail** button will not open your email client. For instructions on registering a default email client, refer to your browser's or operating system's online help.

   Click **Copy to Clipboard** to copy the information to the computer's clipboard. You can paste this information into the Advanced license request and edit it as necessary. For more information, see Using the Advanced License Request Option.

Using the Advanced License Request Option

The advanced license request option lets you request a license by manually typing the information. Make sure to include all of the information required as described in Requesting a License. The following is a suggested format for the advanced option:

```
Request type:New License
License to: <company name>
License type: Permanent
Expiration date:
Hosts: bee01-xp.company.local
Source Types: Oracle
Target Types: SQLServer
```

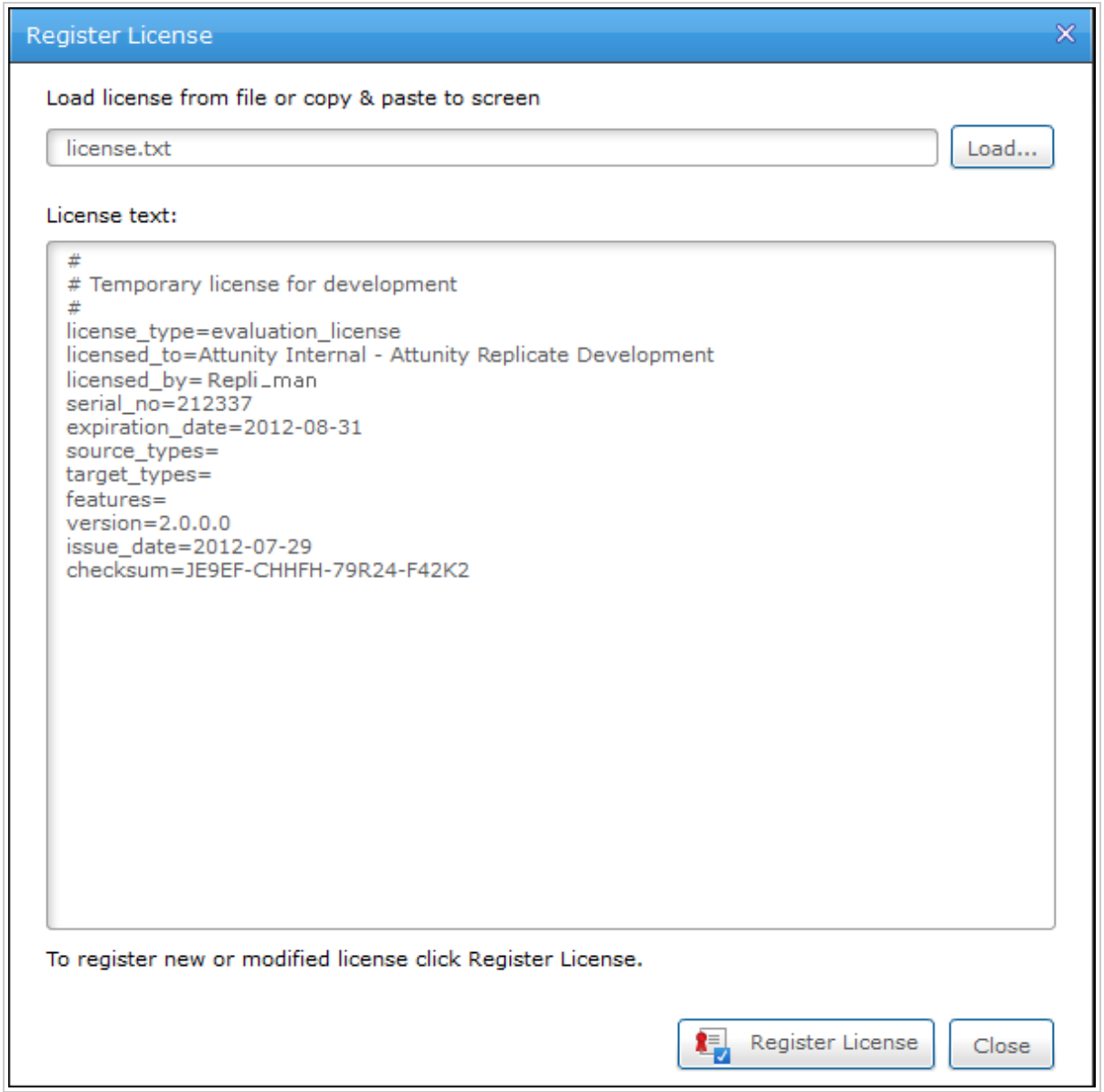
Registering a License

You must have a valid license to work with Attunity Replicate. If you did not receive a license.txt file, you can request a license using the procedure described in Requesting a License. Once you receive the license, you must register it to work with Attunity Replicate.

To register a license:

1. Copy the `license.txt` file to your computer or any computer in your network you have access to.
2. From the Server page, click **License**.
3. At the top of the **License** tab, click **Register License**.

   The **Register License** dialog box opens.



4. Click **Load** and browse to find and select the license file.

   The license text is displayed in the dialog box as shown above. Check to be sure that the details are correct.
5. Click **Register License** to register the license. A message indicating the license was registered successfully is displayed.

> **Note** A message is displayed at the top of the Attunity Replicate Console that indicates that you have a valid license and when it expires. If the license is expired or invalid, the message indicates this.
>
> You can also click on this message link to request, register, or view license information.

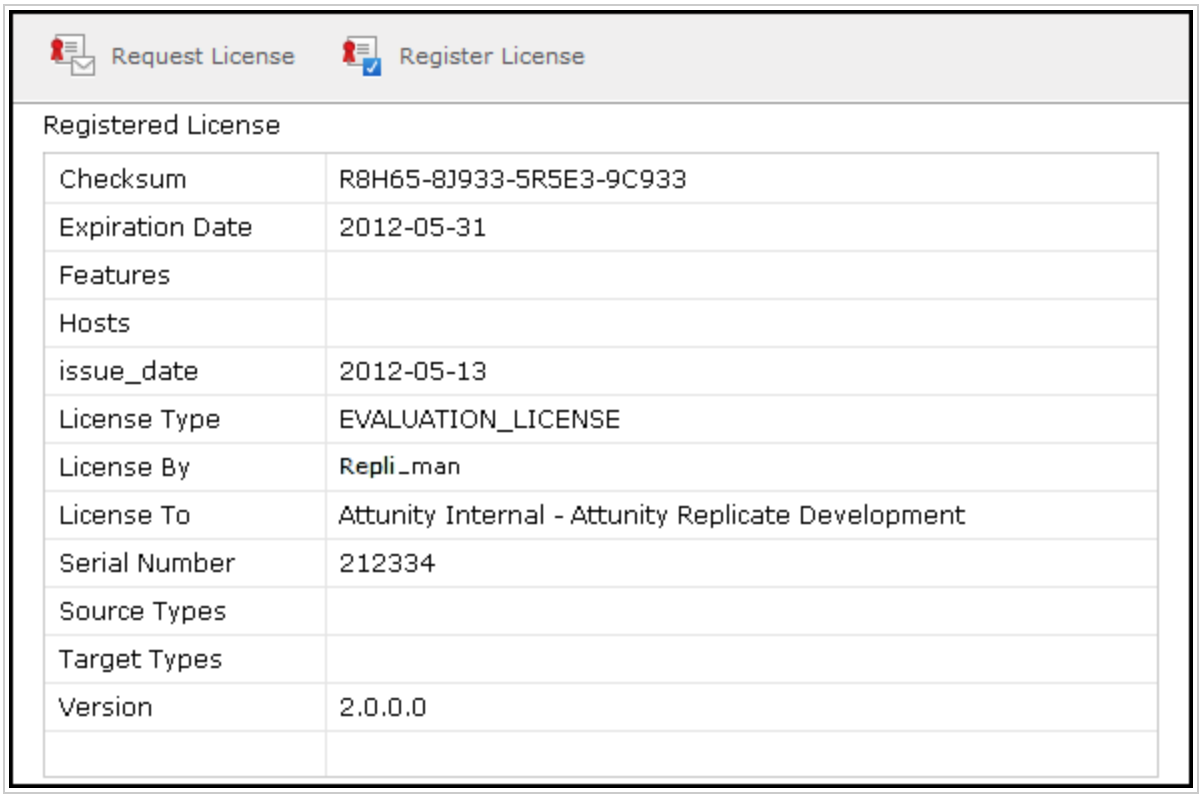
Viewing a License

You can view the license information in the Attunity Replicate Console at any time.

To view the license information:

» From the Server page, click **License**.

The **License** tab is displayed. All of the license information is displayed in the **License** tab.



Global Error Handling

You can configure how Attunity Replicate responds to specific types of errors. You can define error handling on the task level or the server level. The configurations you make in the Server Settings affect all tasks created for this instance of Attunity Replicate unless you define a task to use the definitions you create for that task. For information on how to configure error handling for a specific task, see Error Handling in the Customizing Tasks chapter.

To open the Error Handling page:

From the Server view, click **Global Error Handling** from the menu list on the left.

The following tabs are available:

- **Environmental Errors**: An error that is caused by an environmental problem in the source or target endpoint or on the network. Environmental errors can be restarted.

  The information you enter in this tab is the same as the information you enter in the **Environmental Errors** tab for tasks. For information about the options available in this tab, see Environmental Errors.
- **Data Error**: An error related to data processing at the record level.

  The information you enter in this tab is the same as the information you enter in the **Data Error** tab for tasks. For information about the options available in this tab, see Data Errors in the Customizing Tasks chapter.
- **Table Error**: An error in processing data or metadata for a specific table. This only includes general table data and not an error that relates to a specific record.

  The information you enter in this tab is the same as the information you enter in the **Table Error** tab for tasks. For information about the options available in this tab, see Table Errors in the Customizing Tasks chapter.
- **Apply Conflicts**: Errors that occur when the target endpoint is not synchronized with the source endpoint when processing changes. This can cause duplicate key errors on INSERT operations or zero rows affected on UPDATE/DELETE operations.

  The information you enter in this tab is the same as the information you enter in the **Apply Conflicts** tab for tasks. For information about the options available in this tab, see Apply Conflicts in the Customizing Tasks chapter.

For additional information about errors handling in Attunity Replicate, see the Error and Crash Handling appendix.

Logging

The following topics describe the server logging management options:

- Setting Logging Levels for the Server and File Transfer Service
- Setting Automatic Roll Over and Cleanup

Setting Logging Levels for the Server and File Transfer Service

You set the logging level for Replicate Server logs and File Transfer Service logs in **Server** view. The level you set determines what information is written to the logs. The Server logs provide information about the Attunity Replicate Server instance you are working with as opposed to individual tasks. For information on configuring the task logs, see Setting the Task Logging Level.

The following logging levels are available, ordered from the lowest level to the highest:

1. Errors
2. Warnings
3. Info
4. Trace
5. Verbose

The higher levels always include the messages from the lower levels. Therefore, if you select **Error**, only error messages are written to the log. However, if you select **Info**, informational messages, warnings, and error messages are included. Selecting **Verbose** writes all possible messages to the log.

You can set a global logging level for all components or you can set a separate logging level for each component.

To set the logging levels:

1. On the left side of the **Server** view, click **Logging** and then click the **Server Logging Levels** or **File Transfer Service Logging Levels** sub-tab as required.

   The **Component Logging Level** sliders are displayed.
2. To set a global logging level, move the top slider (the slider with the labels) to the log level you want. Note that all of the sliders for the individual modules move to the same position that you set in the main slider.
3. Make any changes to the sliders for the individual modules. This is optional. Note that if you change the main slider, all of the individual sliders are reset to the new position. If you want to maintain a different logging level for a specific module, you need to reset it.
4. Click **Save** at the top of the screen. Changes to the logging level take place immediately. There is no need to restart the Attunity Replicate Server.

> **Note** Click **Save** to save the information for *all* settings, not only for the logging settings. If you made changes that you do not want to keep, click **Discard Changes** to discard all changes before you make changes to any of the other settings.

Setting Automatic Roll Over and Cleanup

In the **Log File Management** sub-tab, you can define when Replicate should roll over the server log files and when to clean old log files from the system. The following options are available:

- **Enable automatic roll over**: Select this check box if you want to determine when to stop logging to a specific file and begin to log to a new log file. The current log file is called `repserv` and saved (older) log files have the file name `repserv_xxxxxxxxxxxx` where `xxxxxxxxxxxx` represents a 12-digit timestamp.
  - **Roll over the log if the log file is older than (days)**: Use the counter or type in the number of days that to use a specific log file. After the specified number of days the old log is saved with a timestamp appended to its name and a new log file is started.

    The default value is 7 days.
  - **Roll over the log if the log file is larger than (MB)**: Use the counter or type in the maximum amount of megabytes for a specific log file. When the log file reaches the specified size, the old log is saved with a timestamp appended to its name and a new log file is started.

    The default value is 100 megabytes.

    Note If you edit this setting while tasks are running, the new setting will not affect the task log files until the tasks are stopped and then resumed. The server log files are not affected by this limitation.

    Note The scheduled job that checks the log size runs every five minutes. Consequently, the actual size of the log when rolled over might be larger than specified.
- **Enable automatic cleanup:** Select this check box if you want to define the maximum number of days or what size a log file can be before it is deleted.
  - **Delete log files that are older than (days)**: Use the counter or type in the maximum number of days to keep a log file. Log files that are older than the number of days entered are automatically deleted from the system. For example, if you enter **4**, then all log files are deleted on the fifth day.

    The default value is 45 days.

Viewing and Downloading Log Files

You can view log files and download them if necessary.

To view or download the log files:

1. Select the **Server Logging Levels** or **File Transfer Service Logging Level** sub-tab as required.

2. Click the **Log Viewer** toolbar button.

   The **Log Viewer** window opens.
3. Continue from step 4 in Viewing and Downloading the Task Log Files.

Manually Rolling Over the Log Files

You can manually roll over the Replicate Server or File Transfer Service log files if necessary. This lets you stop logging to the current log file and begin to log to a new log file. The currently open log file does not have a timestamp. The name of saved (older) log files is appended with a 12-digit timestamp.

To roll over the log files:

1. Select the **Server Logging Levels** or **File Transfer Service Logging Level** sub-tab as required.
2. Click the **View Logs** toolbar button.

   The **Log Viewer** window opens.
3. Select the log file without a timestamp and then click the **Roll Log File** toolbar button.

Deleting Server, Task and FTS Log Files

You can manually delete task, server, and File Transfer Service log files older than the specified number of days.

To delete the log files:

1. Select the **Logging** tab.
2. Click the **Delete Logs** toolbar button.

   The **Delete Logs** window opens.
3. Select which logs to delete and, for each log, optionally change the default number of days (45).
4. Click **Delete**.

   Selected logs older than the specified number of days will be immediately deleted.

File Transfer Service

The Attunity File Transfer Service (FTS) is a robust and reliable file transfer engine designed to efficiently transfer files over the WAN. This can dramatically improve transfer speeds when the source endpoint and the target endpoint are located on different LANs.

How it Works

A solution using FTS consists of two Attunity Replicate Servers: A local Attunity Replicate Server installed on the source endpoint LAN and a remote Attunity Replicate Server

installed on the target endpoint LAN.

A local task on the local server is defined from the source endpoint to a File Channel target. A remote task on the remote Attunity Replicate Server is defined from a File Channel source to the target endpoint.

The FTS runs on the remote Attunity Replicate Server only and transfers the File Channel files from the storage location defined in the local task to the storage location defined in the remote task.

Upon file transfer, and before Compression and Encryption, large files are split into smaller blocks which form recoverable transport units, and small files are merged into bigger blocks to be sent at the same time. The blocks are then transferred and reconstructed into File Channel files when received by the FTS server.

For information on setting up a File Channel source or target to use FTS, see Using Advanced Properties for a File-Channel Source and Setting Advanced Connection Properties respectively.

Compression

File Channel files are compressed upon sending using GZIP. You can disable the compression and control the compression level.

Encryption

After compression, File Channel files are encrypted using a randomly generated AES-256 session key. The session key is exchanged between the client and server using the Diffie-Hellman key exchange protocol which is authenticated using a secret key that is shared between the client and the server.

Note The File Transfer Service should be configured on the remote Attunity Replicate Server *only*.

Defining a File Transfer Service

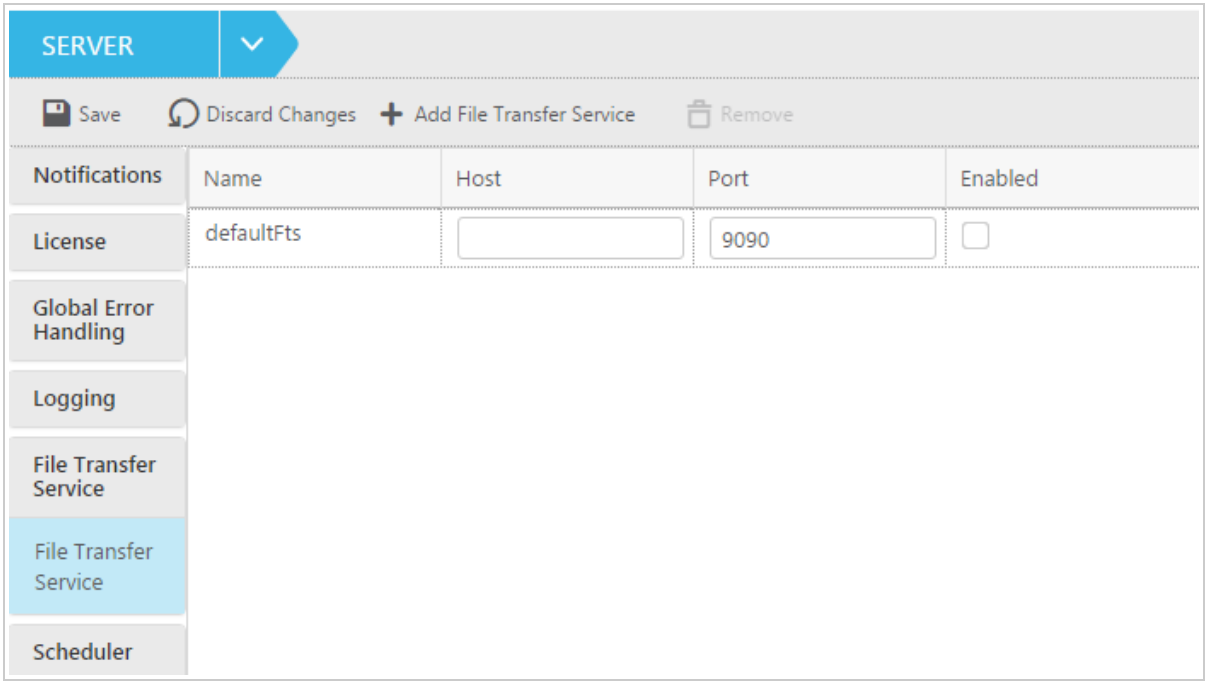
Define a File Transfer Service as described below.

To add a File Transfer Service:

1. Switch to **Server** view as described in Server View.
2. In the left side of the **Server** view, click **File Transfer Service**.
   The File Transfer Service list is displayed.
3. In the Actions toolbar, click **Add File Transfer Service**.
   The **Add File Transfer Service** window opens.
4. Edit the values in the **Name**, **Host** and **Port** columns as follows:

- **Name:** The name of the File Transfer Service.
- **Host:** The host name or IP address of machine on which the remote Attunity Replicate Server is installed. The default is 0.0.0.0 (all interfaces). If the server has multiple NICs (Network Interface Cards), you can define a different File Transfer Service for each card.
- **Port:** The port through which the File Channel files are received.
- **Enabled**: select the check box to enable the File Transfer Service.

5. Click **Save** to save your settings.



Editing a File Transfer Service

You can edit a File Transfer Service as described below.

To edit a File Transfer Service:

1. Select the File Transfer Service you want to edit.
2. Edit the values in the **Name**, **Host** and **Port** columns as follows:
   a. Click the cell to make it editable.
   b. Change the value as required and then click **Save**.

Note When you edit a File Transfer Service, make sure that any File Channel targets configured to use the File Transfer Service are also updated accordingly. For more information on File Channel Targets, see Setting General Connection Properties.

Deleting a File Transfer Service

You can delete File Transfer Services that you no longer want to use.

To delete a File Transfer Service:

1. In the File Transfer Services List, select the item you want to delete.
2. Click the **Remove** toolbar button.

Scheduling Jobs

Use the Attunity Replicate Scheduler to schedule a one-time job or a recurrent job for specific task operations. A job is essentially an operation that can be scheduled to occur once, daily, weekly or monthly.

The following operations can be scheduled:

- Run/Resume a task
- Stop a task
- Reload a task

To schedule a new job:

1. Switch to **Server** view as described in Server View.
2. In the left side of the **Server** view, click the **Scheduler** tab.

   The **Scheduler** tab consists of two sub-tabs: **Scheduled Jobs** and **Executed Jobs**. The **Scheduled Jobs** tab contains a list of jobs that are scheduled to run periodically or once only while the **Executed Jobs** tab contains a list of jobs that have already run.

   Note The **Executed Jobs** tab will only show executed jobs that were scheduled to run once only. In other words, jobs scheduled to run periodically (e.g. Daily, Weekly, Monthly) will not be shown.

3. Click the **New Scheduled Job** toolbar button.

   The **New Scheduled Job** window opens.
4. Specify a **Job Name** and then, from the **Select scheduled job type** drop-down list, select one of the following:
   - **Run task** to run or resume the task(s) at the scheduled time.

     Note For Full Load only tasks, it is preferable to select **Reload target** rather than **Run task** when the scheduling is set to Daily, Weekly or Monthly. This will update the table's data whereas **Run task** will replace the existing table.

   - **Stop task**
   - **Reload target**

Note Selecting **Reload target** will execute the task according to the task's replication settings. For example, if the task's **Full Load** and **Apply Changes** options are enabled, **Reload target** will reload the target tables and apply any subsequent changes.

5. Select one of the following time conventions:
   - **Use server local time** - When this option is selected (the default), the job will run when the specified time is reached *in the server's location*.

     See also: Impact of DST Change on Attunity Replicate.
   - **Use universal time (UTC)** - When this option is selected, the job will run at the specified UTC time. So, for example, if the server is located in a UTC + 2 timezone, a job scheduled to run at 13:00 UTC time will actually run at 15:00 local server time. Scheduling a job to run in UTC mode may be useful if you need tasks on several Replicate servers (located in different timezones) to run concurrently.

Note For reference, both the server local time and the UTC time are displayed to the right of the **Scheduled Time** heading.

6. Select and define one of following scheduling options:
   - Once (Run the job once on the specified day and at the specified time)
   - Daily - (Run the job every day at the specified time)
   - Weekly - (Run the job on the specified days and at the specified time)
   - Monthly - (Run the job on the specified day of the month)

Note To run the job on the last day of evey month, select **Last day of every month** from the **Day of month** drop-down list.

   Optionally, select the **Run job as soon as possible after scheduled time is missed** check box. If the Attunity Replicate Server is offline for whatever reason (e.g. for maintenance), any jobs that were scheduled to run during that time, will be submitted after the machine is brought back online (at the earliest opportunity).
7. For the **Apply to tasks** option, select which tasks to schedule. Select either **All tasks** to apply the job to all current and future tasks or **Selected tasks** to apply the job to specific tasks. If you choose **Selected tasks,** a list of currently defined tasks is displayed. Select which tasks to apply the job to.
8. Click **OK** to save your settings.

To enable or disable a scheduled job:

- In the **Scheduled Jobs** tab, select or clear the check box in the **Enabled** column as required.

To edit a scheduled job:

1. Select the job in the **Scheduled Jobs** or **Executed Jobs** list.
2. Click the **Open** toolbar button and edit the job as required.

To delete a scheduled job:

1. Select the job in the **Scheduled Jobs** or **Executed Jobs** list.
2. Click the **Delete** toolbar button.

User Permissions

You can grant Attunity Replicate users different permissions according to the tasks you want them to perform. Four predefined "roles" are available: Admin, Designer, Operator and Viewer. Each role has its own set of permissions, which are described in the following table.

Default User Permissions According to Roles

Permissions	Admin	Designer	Operator	Viewer
Perform Monitor tasks (in **Monitor** view)	Yes	Yes	Yes	Yes
View and download log files	Yes	Yes	Yes	No
Perform runtime operations (such as start, stop, or reload targets)	Yes	Yes	Yes	No
Create and design tasks	Yes	Yes	No	No
Delete tasks	Yes	Yes	No	No
Manage endpoints	Yes	Yes	No	No
Edit the following server settings: Notifications, license registration, scheduled jobs, and executed jobs.	Yes	Yes	Yes	No
Edit the following server settings: Mail server settings, default notification recipients, global error handling, logging, file transfer service, user permissions, and resource control.	Yes	No	No	No

Note The user under whose account Attunity Replicate is installed will be associated with the Admin role by default.

You can set user permissions using Active Directory Groups or Local Groups. To set user permissions using Active Directory groups, you can either create Active Directory groups with the names listed in the following table or you can create Active Directory groups with

different names. Then, add users to the groups according to the role you want them to perform.

Table 14.7 | Roles and Their Corresponding Active Directory Names

Role	Active Directory Groups
Administrator	AttunityReplicateAdmins
Designer	AttunityReplicateDesigner
Operator	AttunityReplicateOperator
Viewer	AttunityReplicateViewer

For information on encrypting user permissions, see Encrypting the User Permissions File.

Note If you create Active Directory groups with different names, you need to add them to the **User Permissions** window and set their permissions as described in Managing User Permissions.

Managing User Permissions

This section explains how to edit user permissions as well as how to add or remove users and groups.

To edit the user permissions:

1. Switch to **Server** view as described in Server View.
2. In the left side of the **Server** view, click the **User Permissions** tab.
3. Adjust the permission sliders as desired.
4. Click **Save** to save your settings or **Discard** to revert them.

To add new users or groups:

1. Switch to **Server** view as described in Server View.
2. In the left side of the **Server** view, click the **User Permissions** tab.
3. Click the **Add toolbar** button.
   The **Add User/Group** dialog box opens.
4. Select **User** or **Group** as appropriate.
5. Enter the user or group name in the following format:
   For domain users/groups: `domain\group_name` or `domain\user_name`

   Note Active Directory distribution groups are not supported.

   For local users/groups: `computer_name\group_name` or `computer_name\user_name`

Then click **OK**.

The user/group is added to the **User**/**Group** list.

6. Click **Save** to save your settings or **Discard** to revert them.

To remove a user or group:

1. Switch to **Server** view as described in Server View.
2. In the left side of the Server view, click the **User Permissions** tab.
3. Select the user/group you want to remove and then click the **Delete** toolbar button.

   The user/group is deleted.
4. Click **Save** to save your settings or **Discard** to revert them.

Resource Control

You can set high and critical disk space and memory utilization thresholds. Thresholds are calculated as a percentage of total capacity. So, for example, a disk space utilization threshold of 80% would mean that 20% of available disk space remains.After setting the thresholds, you can click the **New Notification** button to define a notification that will be sent whenever a given threshold is exceeded and/or returns to normal.

Disk Space

In the **High Disk Space Utilization Threshold** section, specify the high disk space utilization threshold (in terms of percentage). When the threshold is reached, a notification will be sent (if defined).

In the **Critical Disk Space Utilization Threshold** section, specify the critical disk space utilization threshold (in terms of percentage). When the threshold is reached, all tasks will be stopped and a notification will be sent (if enabled). Replicate will resume the tasks automatically when there is sufficient disk space to do so.

System Memory

Memory utilization is calculated using the following formula (note that "swap file" is used generically to refer to both page file memory on Windows and swap file memory on Linux):

```
(used_swap_file + used_physical_memory) /
(total_swap_file + total_physical_memory) * 100
```

Example:

```
(5 GB + 5 GB) / (10 GB + 10 GB) * 100 = 50%
```

In the **High System Memory Utilization Threshold** section, specify the high system memory utilization threshold (in terms of percentage). When the threshold is reached, a notification will be sent (if defined).

In the **Critical System Memory Utilization Threshold** section, specify the critical system memory utilization threshold (in terms of percentage). When the threshold is

reached, Replicate will start stopping tasks and a notification will be sent (if enabled). The tasks will be resumed automatically when there is sufficient memory to do so.

A | Using Change Tables

You can use Attunity Replicate tasks to save the change events in change tables. This section describes how to use Attunity Replicate with change tables.

In this appendix:

Working with Change Tables

In addition to replicating changes from source endpoint tables to corresponding tables in a target endpoint, Attunity Replicate can also replicate changes to corresponding change tables in the target endpoint. This process occurs simultaneously when applying changes to the target tables. Attunity Replicate lets you determine whether to Replicate the changes to the target only, store changes in the change tables, or both. See Using the Change Table Model below for more information.

The change tables have the same names as the tables that are being replicated, however a suffix is added to each table name. By default the suffix is `__ct`, however you can change the suffix name in the Attunity Replicate Console. For more information on changing the suffix added to the change table names, see Store Changes Settings.

In addition to the selected columns from the source table, the change table also includes special header columns that provide more information on the change the row represents such as the operation, the transaction and the timestamp. This lets you use SQL Query Language to carry out various analyses of the change events, such as fraud detection, trend analysis, triggering of business processes, and Disaster Recovery. For more information about reading the change tables, see Reading the Change Tables.

Using the Change Table Model

When you work with Change Tables you can determine whether to store the changes into the change tables, apply the changes to the target tables, or both store and apply the changes. You determine this when you define the replication task. For more information on this setting, see Store Changes Settings.

In cases where you are both applying and storing the changes, the following is true:

- The target and change tables must be in the same endpoint although they can have different schemas. For example, the Change Tables will contain the metadata headers.
- Changes applied to the Change Table will be handled exactly the same as the changes

performed in the corresponding transaction in the source database. Therefore, when using T**ransactional apply mode** or **Batch optimized apply** mode with the **Preserve transaction consistency** option selected, the changes will be processed as a single transaction.

The exception to this is when an error is encountered and Replicate switches to "one-by-one" apply mode in order to determine which of the Change operations is responsible for the error.

- The same data columns are both applied and stored with the exception of the change header columns, which are only added to the stored change tables.

Reading the Change Tables

You can use the tools for your target endpoint to get information using the metadata in the change tables. This data is defined by the header columns added to the change table schema. For information on these headers, see Change Tables.

Change Tables

For every target table in the replication task, a change table with the corresponding name is maintained by Attunity Replicate in the endpoint with the target tables. For more information, see Working with Change Tables. A change table contains the original table columns, and header columns. The header columns contain a prefix so that the name does not conflict with the source table column names. The default prefix is `header__`. For information on how to change this prefix, see the **Change tables** listing under Metadata in Task Settings. The following table lists the default change table header columns.

Table A.1 | Change Table Headers

Column Name	Type	Description
`[header_ _]change_seq`	varchar (35)	A monotonically increasing change sequencer that is common to all change tables of a task. The change sequence has the following format: YYYYMMDDHHmmSShhxxxxxxxxxxxxxxxxxxx Where: » YYYY is the four-digit year (such as 2012) » MM is the two-digit month (range from 01-12) » HH is the hour in the day (range from 00-23) » mm is the minute in the hour (range from 00-59) » SS is the second in the minute (range from 00-59) » hh is the hundredth of the second (range from 00-99)

Table A.1 | Change Table Headers (Cont.)

Column Name	Type	Description
		» xxxxxxxxxxxxxxxxxxx is a 19-digit, zero prefixed change number (global per task). The time part usually refers to the commit time of the transaction that includes the change record. Attunity Replicate contains logic that maintains the monotonicity of the sequence number so modifying or adjusting the endpoint time may result in multiple changes to seem that they are within the same timestamp but with increasing change number. The xxx...xxx is usually the internal change number from the data record except that for BEFORE-IMAGE records it is the same as the change number of the matching UPDATE record (for example, if the change number of BEFORE-IMAGE is 1000 and that of the UPDATE is 1001, then both have 1001). This allows a simple left-outer-join between the table and itself where on the left we scan until the point in time but filter out `operation=before-image`, and on the right we join on the same `change_seq` with the `change_oper` being 'B' .
`[header__]change_oper`	varchar (1)	The operation type. This can be one of the following: » `I`: INSERT » `D`: DELETE » `U`: UPDATE » `B`: Before Image
`[header__]change_mask`	varbinary (128)	The change mask indicates which data columns in the change table are associated with columns that changed in the source table. The bit position in the change mask is based on the column ordinal in the change table. This means that if there are 5 header columns, they occupy bits 0 to 4 and the first data column is bit 5 in the change mask. The change mask is a binary column (a byte array) representing the change mask in little-endian order: **Byte 0** bit7 bit6 bit5 bit4 bit3 bit2 bit1 bit0 **Byte 1** bit15 bit14 bit13 bit12 bit11 bit10 bit9 bit8 In this example, `bit#N` indicates that the change table column of ordinal N relates to a column that changed in the source table. If update mask is 11000 and the column ordinal is 3 the column did not change.

Table A.1 | Change Table Headers (Cont.)

Column Name	Type	Description
		The following describes the bit semantics: » For INSERT records, all non-null columns have the associated bits set. » For DELETE records only primary-key (or unique index) columns have the associated bits set. This allows an applier to construct a DELETE statement without having to find the primary key fields from another source. » For BEFORE-IMAGE records, all bits are clear (the change mask can be empty). » For UPDATE records, each column whose value changed between the BEFORE-IMAGE and the UPDATE will have the associated bit set. For space and processing efficiency, the actual number of bytes stored in the change-mask can be null-trimmed. This means that trailing zeros do not need to be stored. The handling logic should take this into consideration.
[header__] stream_position	varchar (128)	The source CDC stream position.
[header__] operation	varchar (12)	The operation associated with the change record. It can be one of the following: » INSERT » UPDATE » DELETE » BEFOREIMAGE
[header__] transaction_id	varchar (32)	The ID of the transaction that the change record belongs to. The value is a hex-string of the 128-bit transaction ID.
[header__] timestamp	timestamp	The original change UTC timestamp (the value may be approximate). **Note:** With PostgreSQL source, the timestamp is only known after the commit occurs. Therefore, until the changes are committed to the source tables, the default date will be displayed (e.g. 1970-01-01).
[header__] partition_	string	The name of the partition created on the target when Change Data Partitioning is enabled. The partition name consists of

Table A.1 | Change Table Headers (Cont.)

Column Name	Type	Description
name		the partition start and end time. **Example:** 20170313T123000_20170313T170000

Use Example

The following SQL statement returns a range of changes that starts after the last handled event (of change sequence "20120723144522110000000000000901203") and until the event committed on 23-Jul-2012 at 23:00:00.00. For update we also get the before image value (here for the salary before and after).

```
SELECT CHANGE.[header__change_seq]
      ,CHANGE.[header__stream_position]
      ,CHANGE.[header__operation]
      ,CHANGE.[header__transaction_id]
      ,CHANGE.[header__timestamp]
      ,CHANGE.[EMPLOYEE_ID]
      ,CHANGE.[FIRST_NAME]
      ,CHANGE.[LAST_NAME]
      ,CHANGE.[SALARY]
      ,BI.[SALARY]
   FROM [Replication].[HR].[EMPLOYEES_ct] CHANGE LEFT OUTER JOIN
      [Replication].[HR].[EMPLOYEES_ct] BI  ON
         BI.[header__change_seq] = CHANGE.[header$__change_seq] AND
         BI.[header__change_oper] = 'B'
   WHERE CHANGE.header__oper <> 'B' AND
         CHANGE.[header__stream_position] >
         '20120723144522110000000000000901203' AND
         CHANGE.[header__stream_position] <= '20120723230000000Z' AND
   ORDER BY
         CHANGE.[header__stream_position], CHANGE.[header$__stream_oper]
```

B | Using an Audit Table

When defining a replication task, you can choose to store changes to all tables in a single audit table (located in the target endpoint). The changes can then be pushed from a single stream to any queue (JMS for example).

The following table describes the structure of the Audit table.

Table B.1 | Audit Table Headers

Column Name	Type	Description
`task_name`	varchar (128)	The name of the Attunity Replicate replication task.
`change_seq`	bigint	A monotonically increasing change sequencer. The change sequence has the following format: YYYYMMDDHHmmSShhxxxxxxxxxxxxxxxxxxx Where: YYYY is the four-digit year (such as 2012) MM is the two-digit month (range from 01-12) HH is the hour in the day (range from 00-23) mm is the minute in the hour (range from 00-59) SS is the second in the minute (range from 00-59) hh is the hundredth of the second (range from 00-99) xxxxxxxxxxxxxxxxxxx is a 19-digit, zero prefixed change number (global per task). The time part usually refers to the commit time of the transaction that includes the change record. Attunity Replicate contains logic that maintains the monotonicity of the sequence number so modifying or adjusting the endpoint time may result in multiple changes appearing as if they are within the same timestamp, but with increasing change number. The xxx...xxx is usually the internal change number from the data record except that for BEFORE-IMAGE records it is the same as the change number of the matching UPDATE record (for example, if the change number of BEFORE-IMAGE is 1000 and that of the UPDATE is 1001, then both have 1001).
`change_oper`	varchar (1)	The operation type. This can be one of the following: `I`: INSERT `D`: DELETE

Table B.1 | Audit Table Headers (Cont.)

Column Name	Type	Description
		`U`: UPDATE `B`: Before Image
`stream_position`	varchar (128)	The source CDC stream position.
`schema_name`	nvarchar (128)	The name of the source table schema.
`table_name`	nvarchar	The name of the source table.
`operation`	varchar (12)	The operation associated with the change record. It can be one of the following: » INSERT » UPDATE » DELETE » BEFOREIMAGE
`transaction_id`	varchar (32)	The ID of the transaction that the change record belongs to. The value is a hex-string of the 128-bit transaction ID.
`timestamp`	timestamp	The original change timestamp (the value may be approximate).
`change_record` **Note** The data is stored in JSON format.	nclob	The new data. **Note** LOB columns with unlimited size are not supported in the `change_record` field. The other fields will be recorded but the LOB will have a NULL value.
`bu_change_record` **Note** The data is stored in JSON format.	nclob	The data before the update. **Note** LOB columns with unlimited size are not supported in the `bu_change_record` field. The other fields will be recorded but the LOB will have a NULL value.

For more information on storing changes in an audit table, see Store Changes Settings.

C | Error and Crash Handling

Attunity Replicate must handle different types of errors during its operation. The way the system should respond to these errors depends on several aspects, including the component where the error occurred, the type of error, and the scope of the error. Because different sites may have different requirements for error behavior, Attunity Replicate lets you configure the error handling.

You can also add an environment variable that instructs Replicate to create dump files in the event of a crash. The dump files can then be used by Attunity Support to troubleshoot the cause of the crash.

In this appendix:

Error Types

This topic provides information on the different types of errors that you may encounter using Attunity Replicate. The following is a list of the error types:

- Environmental Errors: An error that is caused by an environmental problem in the source or target endpoint or on the network. Some examples of environmental errors are loss of communication with the source or target endpoint, restarting an endpoint, or network problems.
- Data Errors: An error related to data processing at the record level. Some examples of data errors are conversion errors, errors in transformations, or bad data.
- Table Errors: An error in processing data or metadata for a specific table. This only includes general table data and not an error that relates to a specific record.
- Apply Errors (conflicts): Errors that occur when the target endpoint is not synchronized with the source endpoint when processing changes. This can cause duplicate key errors on INSERT operations or zero rows affected on UPDATE/DELETE operations.
- Fatal Errors: An error that is not related to a specific table but is not environmental. Some examples of fatal errors are incorrect endpoint configuration or incorrect password information.
- Abnormal Termination: An error that is returned when the task terminates for an abnormal or unknown reason.

Error Handling Properties

This topic describes the properties that you can set to determine how errors are handled. For information on how to configure error handling behavior in Attunity Replicate, see Error Handling in Customizing Tasks or Global Error Handling in Attunity Replicate Server Settings.

See the following topics for information about configuring the error behavior for each type of error:

- Environmental Errors
- Data Errors
- Table Errors
- Apply Errors
- Fatal Errors
- Abnormal Termination

Environmental Errors

When an environmental error occurs, the system will attempt to restart the task if it stops. You can define exactly how and when the system will try to restart a task.

The following table describes the properties used to determine how to handle environmental errors.

Table C.1 | Properties to Define Environmental Error Handling

Name	Description	Default
RETRY_ COUNT	The maximum number of attempts made to restart a task when an environmental error occurs. After the system attempts to restart the task the designated number of times, the task is stopped and manual intervention is required. Enter -1 to continuously attempt to restart a task. Enter 0 to never attempt to restart a task.	-1
RETRY_ INTERVAL	The number of seconds that the system waits between attempts to restart a task.	5
RETRY_ THROTTLING	When active, the interval between attempts to restart a task gets longer each time.	Y
RETRY_ THROTTLING_ MAX	The maximum number of seconds to wait between attempts to restart a task if RETRY_THROTTLING is on.	1800

Data Errors

When a data error occurs (to a specific record), you can define what you want the task to do and how to handle the error.

The following table describes the properties used to determine how to handle data errors.

Table C.2 | Properties to Define Data Error Handling

Name	Description	Default
DATA_ TRUNCATION_ ERROR_ POLICY	Determine what to do when data is truncated in one or more records on the target. You can: IGNORE_RECORD: The task continues and the data for that record is ignored. The error counter for the DATA_ERROR_ ESCALATION_COUNT property is increased so that if you set a limit on errors for a table, this error will count toward this limit. LOG_ERROR: The task continues and the error is written to the task log. SUSPEND_TABLE: The task continues but data from the table with the error record is moved into an error state and its data is not replicated. STOP_TASK: The task is stopped and manual intervention is required.	LOG_ ERROR
DATA_ERROR_ POLICY	Determine what to do when an error occurs to the data from a specific record being replicated. You can: IGNORE_RECORD: The task continues and the data for that record is ignored. The error counter for the DATA_ERROR_ ESCALATION_COUNT property is increased so that if you set a limit on errors for a table, this error will count toward this limit. LOG_ERROR: The task continues and the error is written to the task log. SUSPEND_TABLE: The task continues but data from the table with the error record is moved into an error state and its data is not replicated. STOP_TASK: The task is stopped and manual intervention is required.	LOG_ ERROR
DATA_ERROR_ ESCALATION_ COUNT	Set the maximum number of errors that can occur to the data for a specific record. When this number is reached, the data for the table that contains the error record is handled according to the policy set in the DATA_ERROR_ ESCALATION_ACTION.	0, no escal- ation

Table C.2 | Properties to Define Data Error Handling (Cont.)

Name	Description	Default
DATA_ERROR_ESCALATION_ACTION	Determines the action to carry out when the maximum number of errors (set in the DATA_ERROR_ESCALATION_COUNT) is reached. You can select one of the following options: LOG_ERROR: The task continues and the error is written to the task log. SUSPEND_TABLE: The task continues but data from the table with the error record is moved into an error state and its data is not replicated. STOP_TASK: The task is stopped and manual intervention is required.	SUSPEND_TABLE

Table Errors

The following table describes the properties used to determine how to handle errors that occur to the general data or metadata for a specific table. It does not affect the error policy for specific records in the table.

Table C.3 | Properties to Define Environmental Error Handling

Name	Description	Default
TABLE_ERROR_POLICY	Determine what to do when an error occurs to the general table data being replicated. You can: SUSPEND_TABLE: The task continues but data from the table with the error record is moved into an error state and its data is not replicated. STOP_TASK: The task is stopped and manual intervention is required.	SUSPEND_TABLE
TABLE_ERROR_ESCALATION_COUNT	Set the maximum number of errors that can occur to the general data or metadata for a specific table. When this number is reached, the data for the table is handled according to the policy set in the TABLE_ERROR_ESCALATION_ACTION.	0, no escalation
TABLE_ERROR_ESCALATION_ACTION	Determines the action to carry out when the maximum number of errors (set in the TABLE_ERROR_ESCALATION_COUNT) is reached. You can select one of the following options: STOP_TASK: The task is stopped and manual intervention is required.	STOP_TASK

Apply Errors

The following table describes the properties used to determine how to handle Apply errors. Apply errors occur when the source and target endpoints are not properly synchronized during a change processing operation. This causes the following record-level issues:

- DELETE or UPDATE operations occur with zero rows affected.
- INSERT operation occurs with a duplicate key.

Table C.4 |

Name	Description	Default
APPLY_ DELETE_ CONFLICT_ POLICY	Determine what to do when there is a conflict with DELETE operation. You can: IGNORE_RECORD: The task continues and the data for that record is ignored. The error counter for the APPLY_CONFLICT_ ESCALATION_COUNT property is increased so that if you set a limit on errors for a table, this error will count toward this limit. LOG_ERROR: The task continues and the error is written to the task log. FIX_RECORD: The record is corrected so that it shows the correct DELETE operation.[1] SUSPEND_TABLE: The task continues but data from the table with the error record is moved into an error state and its changes are not processed. STOP_TASK: The task is stopped and manual intervention is required.	IGNORE_ ERROR
APPLY_ INSERT_ CONFLICT_ POLICY	Determine what to do when there is a conflict with an INSERT operation. You can: IGNORE_RECORD: The task continues and the data for that record is ignored. The error counter for the APPLY_CONFLICT_ ESCALATION_COUNT property is increased so that if you set a limit on errors for a table, this error will count toward this limit. LOG_ERROR: The task continues and the error is written to the task log. FIX_RECORD: The record is corrected so that it shows the correct INSERT operation.[1] SUSPEND_TABLE: The task continues but data from the table with the error record is moved into an error state and its	LOG_ ERROR

[1]FIX_RECORD requires full supplemental logging in order to ensure that an UPDATE is not turned into an INSERT. In other cases, FIX_RECORD can cause an async full load of a record similar to the LOB channel.

Table C.4 | (Cont.)

Name	Description	Default
	changes are not processed. STOP_TASK: The task is stopped and manual intervention is required.	
APPLY_ UPDATE_ CONFLICT_ POLICY	Determine what to do when there is a conflict with an UPDATE operation. You can: IGNORE_RECORD: The task continues and the data for that record is ignored. The error counter for the APPLY_CONFLICT_ ESCALATION_COUNT property is increased so that if you set a limit on errors for a table, this error will count toward this limit. LOG_ERROR: The task continues and the error is written to the task log. FIX_RECORD: The record is corrected so that it shows the correct UPDATE operation.[1] SUSPEND_TABLE: The task continues but data from the table with the error record is moved into an error state and its changes are not processed. STOP_TASK: The task is stopped and manual intervention is required.	LOG_ ERROR
APPLY_ CONFLICT_ ESCALATION_ COUNT	Set the maximum number of APPLY conflicts that can occur for a specific table during a change process operation. When this number is reached, the data for the table is handled according to the policy set in the APPLY_CONFLICT_ESCALATION_ ACTION	0, no escalation
APPLY_ CONFLICT_ ESCALATION_ ACTION	Determines the action to carry out when the maximum number of errors (set in the APPLY_CONFLICT_ESCALATION_ COUNT) is reached. You can select one of the following options: LOG_ERROR: The task continues and the error is written to the task log. SUSPEND_TABLE: The task continues but data from the table with the conflict is moved into an error state and its data is not replicated. STOP_TASK: The task is stopped and manual intervention is required.	LOG_ ERROR
APPLY_ CONFLICT_	Determine whether to load the conflicting data when carrying out a full-load operation after the change processing is com-	Conflicts are

Table C.4 | (Cont.)

Name	Description	Default
DURING_ FULL_LOAD	plete.	ignored

Fatal Errors

There is no manual configuration for handling fatal errors. In case of a fatal error, the task is stopped and manual intervention is required.

Abnormal Termination

There is no manual configuration for handling fatal errors. In case of abnormal termination, Attunity Replicate attempts to restart the task five times. If the task does not successfully restart for at least 15 minutes, it is stopped and manual intervention is required.

Creating Dump Files

Dump files are created when a component crashes and creates an exception code. They are useful for troubleshooting such crashes as they states at exactly what line or area of memory the crash occurred.

To create a dump file:

1. Create a new environment variable with the following name and value:

   Variable: `AREP_CRASH_DUMP_TYPE`

   Possible values: `FULL`, `MINI` or `NONE`
2. Restart both Replicate services.
3. This step is optional. If `AREP_CRASH_DUMP_TYPE` is set to `FULL` or `MINI`, navigate to the following registry key:

   HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\Windows Error Reporting\LocalDumps\repctl.exe

   Verify that the data value for the **DumpType** Binary Value is either 1 (`MINI`) or 2 (`FULL`).
4. After a crash occurs the dump files will be created in the following directory:

   `<Product_Dir>\minidumps`

   Note When `AREP_CRASH_DUMP_TYPE` is set to `NONE`, no dump files will be created in the event of a crash.

D | Pivotal Greenplum Prerequisites for Attunity Replicate

This section specifies the requirements for working with a Pivotal Greenplum database or endpoints with Attunity Replicate.

In this appendix:

Required Pivotal Greenplum Software Environments

You can use the Pivotal Greenplum database with Attunity Replicate on either a Windows or Linux computer. The following sections describe the prerequisites necessary to prepare your environment to work with Attunity Replicate and a Pivotal Greenplum database.

- Windows Pivotal Greenplum Required Software
- Linux Pivotal Greenplum Required Software

Windows Pivotal Greenplum Required Software

You must install the following on the same computer where the Attunity Replicate Server is installed:

- Greenplum Client Tools 4.2.x
- Greenplum Connectivity Tools 4.2.x
- Greenplum Loaders 4.2.x (This will install the gpfdist program)
- Greenplum DataDirect ODBC Drivers Version 7.x (only)

  You can download the ODBC drivers from emc.subscribenet.com.

Linux Pivotal Greenplum Required Software

You must make sure to configure the Linux computer as follows:

- Greenplum Connectivity Tools 4.2.1.0. Unzip and install all of the components.
- Install and configure **UnixODBC**.

- Ensure that the following Pivotal Greenplum environment scripts are executed when logging to the account where Attunity Replicate is run:
  - greenplum_connectivity_path.sh
  - greenplum_clients_path.sh
  - greenplum_loaders_path.sh
- Make sure that port 8080 is open.

Required Pivotal Greenplum Configuration and Environment

Attunity Replicate relies on the proper functioning of Pivotal Greenplum's gpfdist program on the computer with Attunity Replicate (the local computer). gpfdist is a simple Web server program with special performance customizing for concurrent access from Pivotal Greenplum database segment nodes to data files on the local computer.

Because gpfdist is a Web server program and because it needs to be accessible from the Pivotal Greenplum database segment nodes, there are some networking configuration settings that must be in place to allow for this access. This is documented in the *EMC Greenplum database Administration Guide*.

The following sections provide a simple test that verifies the proper configuration of the Pivotal Greenplum database and the local software installation. You can run this test before installing any Attunity software on your local computer. This test must be completed successfully to ensure that Attunity Replicate can work with a Pivotal Greenplum database.

- Collect Connection Information
- Create a Test Input File
- Create an SQL Script File
- Start gpfdist
- Run the SQL Script

Collect Connection Information

The following is the information required for this test. You should write down the information for your system to use later in this test.

- *<<Pivotal Greenplum-host>>*: The Pivotal Greenplum database host name (master).
- *<<Pivotal Greenplum-port>>*: The Pivotal Greenplum database port number (master). In many cases the port number is 5432
- *<<Pivotal Greenplum-user>>*: The username that is used to connect to the Pivotal Greenplum database. This user must have permission to create an external table.
- *<<Pivotal Greenplum-password>>*: The password of the selected Pivotal Greenplum user.

- <<*Pivotal Greenplum-database*>>: The Pivotal Greenplum database name where the test is created.
- <<*Pivotal Greenplum-host*>>: The local computer name as seen from the Pivotal Greenplum segment nodes.
- <<*gpfdist-port*>>: The port number where the gpfdist program is listening.

Create a Test Input File

Create a text file called `greenplum_test123.txt` with the following content:

```
Scott,Greenplum
Tiger,Woods
```

Create an SQL Script File

Create a text file called `greenplum_test123.sql` with the following content, replacing the <<.fgf.>> tokens with the information for your system that you wrote down in step 1:

```
CREATE EXTERNAL TABLE greenplum_test123 ( name text, descr text )
   LOCATION ('gpfdist://<<gpfdist-host>>:<<gpfdist-port>>/greenplum_
   test123.txt')
   FORMAT 'TEXT' (DELIMITER ',');
SELECT * FROM greenplum_test123;
DROP EXTERNAL TABLE greenplum_test123;
```

Start gpfdist

Open a command shell and change the current directory to the directory where you created the Test Input File and the SQL Script File. Then start the gpfdist program with the following command:

```
$ gpfdist -v -p <<gpfdist-port>> -d .
```

Run the SQL Script

Run the SQL script that you created with the following command:

```
$ psql -d <<Pivotal Greenplum-database>> -h <<Pivotal Greenplum-host>> -p
<<Pivotal Greenplum-port>> -U <<Pivotal Greenplum-user>> -f greenplum_
test123.sql
```

If the script runs successfully, the following is displayed:

```
CREATE EXTERNAL TABLE
name | descr
-------+-----------
Scott | Pivotal Greenplum
Tiger | Woods
```

```
(2 rows)
DROP EXTERNAL TABLE
```

If the script is not successful, you will get an output that is similar to the following example:

```
CREATE EXTERNAL TABLE
psql:greenplum_test123.sql:4: ERROR: connection with gpfdist failed for
gpfdist://atturepl:18080/greenplum_test123.txt. effective url:
http://192.168.165.12:
18080/greenplum_test123.txt. error code = 110 (Connection timed out) (seg0
slice1 greenplum421.acme.local:40000 pid=31364)
DROP EXTERNAL TABLE
```

In the example above, the problem was that a firewall on the local computer prevented the Pivotal Greenplum database segment from reaching the local gpfdist instance.

Any error that occurs must be resolved before using Attunity Replicate with EMC Pivotal Greenplum. Once you resolve the error, you should run this test again to ensure that you can work with the Pivotal Greenplum database.

For information about what to do if this test fails, see Troubleshooting gpfdist Issues.

Troubleshooting gpfdist Issues

If the test described in Required Pivotal Greenplum Configuration and Environment fails, you should carry out the following checks:

- Did gpfdist start on the correct port or protocol?
- Can Pivotal Greenplum reach gpfdist?

Did gpfdist start on the correct port or protocol?

To check whether gpfdist is listening on the correct port or protocol, enter the following command:

For Windows:

```
$ netstat -a -n | find "<<gpfdist-port>>"
```

For Linux:

```
$ netstat -a -n | grep "<<gpfdist-port>>"
```

The following is the output that you should get if gpfdist started on port 8080 on Windows:

```
$ netstat -a -n | find "8080"
TCP 0.0.0.0:8080 0.0.0.0:0 LISTENING
TCP [::]:8080 [::]:0 LISTENING
```

This indicates that gpfdist is listening on any network interface on both IPv4 (0.0.0.0:8080) and IPv6 ([::]:8080).

If only the IPv6 line is shown in most cases there is a local networking configuration problem. Pivotal Greenplum's recommendation in this case is to disable IPv6 on the local computer (see the Microsoft knowledge base article on how to carry this out at http://support.microsoft.com/kb/929852).

Can Pivotal Greenplum reach gpfdist?

For gpfdist to work, all Pivotal Greenplum database segments must be able to communicate with the local machine (the **ETL machine** as described by Pivotal Greenplum) using the HTTP protocol.

While gpfdist is running, run the following command from each one of the Pivotal Greenplum database segment nodes to ensure that all segments can access gpfdist:

```
$ wget http?//<<Pivotal Greenplum-host>>?<<Pivotal Greenplum-port>>/gpfdist/status
```

This should return a status page with the following or similar content:

```
read_bytes 0
total_bytes 0
total_sessions 0
```

When carrying out the network checks from the Pivotal Greenplum internal master node, there is usually a file called `seg_host` that contains a list of the Pivotal Greenplum database segment node names. If this file exists, you can check access from all segment nodes using a single Pivotal Greenplum `gpssh` command:

```
$ gpssh -f seg_host
=> wget http?//<<Pivotal Greenplum-host>>?<<Pivotal Greenplum-port>>/gpfdist/status
```

If this check fails, a network or system manager must change the network or system configuration so that the check succeeds.

E | Setting up Attunity Replicate in a Cluster Environment

This section describes how to set up Attunity Replicate in Windows and Linux clustering environments.

In this appendix:

Setting up Attunity Replicate in a Windows Server Cluster (HA)

This section describes how to set up Attunity Replicate in a Windows Server 2008 or Windows Server 2012 Cluster environment. The following are the required steps for setting up the cluster environment. Please note that these steps provide a general explanation of how to set up the environment to work with Attunity Replicate. If you need directions on how to work with specific programs in Microsoft Windows, see the Microsoft documentation.

Prepare to Set Up the Cluster Environment

You must make sure that your system is ready to use a cluster environment. Carry out the following procedures before you set up your cluster and services.

To prepare to set up the cluster environment:

1. Make sure that the storage, hardware, and hardware settings on each of your servers, are compatible with failover clustering. See the Microsoft documentation for an explanation on how to validate your system.

2. Open the Server Manager console on one of the servers Attunity Replicate is installed. You will need to install Attunity Replicate on this server and all other servers in your cluster environment. See Install Attunity Replicate.

   Note You must make sure that the Failover Clustering snap-in for the Microsoft Management Console is installed on the server where you are configuring the cluster. For more information, see the Microsoft documentation.

3. In the left column of the Server Manager, expand **Features**.
4. Select **the Failover Cluster Manager**.

Create a Failover Cluster and a Service to Use in Your Environment

Carry out this procedure for as many services that you are using.

The first thing you should do is to create a failover cluster that defines the servers or nodes that are included in your cluster environment.You must also create a service to manage the Attunity Replicate application and services.

To create a failover cluster and a service:

1. Right-click the highest node called **Failover Cluster Manager** and select **Create a Cluster**. Follow the directions in the Create Cluster wizard to create a cluster.
2. Expand the cluster you are working with.
3. Right-Click **Services and applications** and point to **More Actions**, then select **Create Empty Service or Application**. Follow the directions in the Wizard to create a new service. If you need additional information on how to create a new service, see the Microsoft documentation.
4. Right-click the service that you created and select **Properties**. In the **Name** field, change the default name to a name that you want to use for this service. The name should have a relevant meaning so you can remember it in the future.

   Note If necessary, you can carry out any of the options in the menu displayed when you right-click the service.

Add Storage

You must define the disk that you use for storage. Use the storage to hold all of the information in the Attunity Replicate data folder.

To add storage:

1. Right-click the Storage node and select **Add a Disk**. This disk will contain the data in the data folder. The data folder is located in the root folder where Attunity Replicate is installed. This disk is bound to the Replicate Services. This means that in case of failover, this disk will be moved to the another cluster running the Attunity Replicate

Services.

2. On the Storage node, you can add the new disk to the failover cluster.
3. Right-Click one of the services in the cluster and select **Add Storage**. Then select the new disk to add this disk to the cluster you are working with.

> **Note** Make sure that there is a disk available on the server where you are adding the disk. Also, to ensure that the system recognizes the disk, make sure that the storage configuration for the disk is correct. The disk you want to use must be configured as a basic disk, not a dynamic disk. For more information, see the Microsoft online help.

Define Client Access

A network name must be added so you can access this service from other machines using the Network Name.

To define client access:

1. Right-click one of the services that you created and point to **Add a resource** then select **Client Access Point**. The services are sub-nodes of Services and applications.
2. Follow the directions in the New Resource Wizard to set the Network Name. The network name must be the same as the service.

   You set the name on the first page (Client Access Point) of the wizard.

Install Attunity Replicate

Perform the following steps to install Attunity Replicate:

1. Run the Attunity Replicate setup wizard on the first node that is part of your cluster (see Create a Failover Cluster and a Service to Use in Your Environment). Install Attunity Replicate to a local folder. The default is:

   `C:\Program Files\Attunity\Replicate`.

   In the setup wizard, specify a shared storage location for the "data" folder; for example, `F:\Replicate\data`.

   For more information, see Attunity Replicate on Windows: Installing, Upgrading and Uninstalling.
2. Browse to the shared storage location, for example `F:\Replicate\data` and delete the **mk.dat** file.

   Then, on the same machine (i.e. the first cluster node), open a command prompt as administrator and change the working directory to the Replicate **bin** directory.

   Run the following command:

   ```
   repctl setmasterkey <your_new_master_key> master_key_scope=1
   ```

   where:

`<your_new_master_key>` is the new master key.

This will create a new **mk.dat** file with a constant scope in the same location.

> **Important:** After the **mk.dat** file is deleted, all stored secrets will no longer be valid (as they were encrypted using the old key). Therefore, after changing the master key, you need to reenter the passwords in all the relevant places. For more information, see Changing and Protecting the Master Key.
>
> In addition, because the Replicate server password is also stored in the **mk.dat** file, you need to reset it as described in Changing the Server Password.

3. Now move your cluster to the next cluster node and install Attunity Replicate as described in Step 1 above. Repeat for the remaining cluster nodes.
4. After Replicate is installed on all nodes, edit the **ServiceConfiguration.xml file** in the "data" folder and change the `Https://` and `Http://` entries to use the Client Access Point name.

Add the Attunity Replicate Services

The two Attunity Replicate services must be added as resources to the service (Windows Server 2008 Cluster) or role (Windows Server 2012 Cluster) you added in the Create a Failover Cluster and a Service to Use in Your Environment procedure.

The Attunity Replicate services are called Attunity Replicate UI Server and Attunity Replicate Server.

To add the Attunity Replicate services:

1. Do one of the following (according to your Windows Server version):
   - **Windows Server 2008 Cluster:** Right-click the service you are working with and point to **Add a resource**. Then select **Generic Service**.
   - **Windows Server 2012 Cluster:** In the left pane of the Failover Cluster Manager, select **Roles**. The available roles will be listed in the right pane of the console. Right-click the role you are working with and point to **Add a resource**. Then select **Generic Service**.
2. In the **Select Service** screen of the New Resource wizard, select **Attunity Replicate UI Server** from the List.
3. Click **Next** and follow the directions in the wizard to create the resource. For information on how to use this wizard, see the Microsoft online help.

> **Note** Attunity Replicate must be installed on the computers where you defined the service for the Attunity Replicate services to be available in the list.

4. Repeat the same steps for the **Attunity Replicate Server**.

Define the Dependencies for Each Service

You should define dependencies for the Attunity Replicate services. This allows the Storage and the Network names to start before the Attunity Replicate services. If these resources do not start up before the services, Attunity Replicate will not start because it will continue to search for the data location.

To define the dependencies:

1. Do one of the following (according to your Windows Server version):
   - **Windows Server 2008 Cluster:** In the left pane of the Failover Cluster Manager, select the **Attunity Replicate UI Server** service.

     The properties for this service are displayed in the center pane.
   - **Windows Server 2012 Cluster:** In the left pane of the Failover Cluster Manager console, select **Roles**. The available roles will be listed in the right pane of the console. Select the role you are working with and then, in the bottom right pane, select the **Resource** tab. From the list of the available roles, select **Attunity Replicate UI Server**.
2. Do one of the following (according to your Windows Server version):
   - **Windows Server 2008 Cluster:** In the **Other Resources** section, double-click the **Attunity Replicate UI Server** service.
   - **Windows Server 2012 Cluster:** Right-click the **Attunity Replicate UI Server** role and select **Properties**.

   The **Attunity Replicate UI Server Properties** dialog box opens.
3. In the **Attunity Replicate UI Server Properties** dialog box, select the **Dependencies** tab.
4. Click **Insert**. A new line is added to the Resource list.
5. In the Resource column, click the arrow and select the Replicate Data storage resource from the list.
6. Click **Insert** and add the Network Name resource (its name should be the same as the cluster name).
7. Repeat the steps for the Attunity Replicate Server service.
8. Start the Services using the Failover Cluster Manager and access the console using the Network name. See Define Client Access for information on how to configure the network name.
9. Register the license. The license should contain all host names of the cluster.

Note To open Attunity Replicate Console, it is recommended to use an address that includes the name or IP address of the cluster machine (as opposed to the specific node name).

Example:

http://cluster_name_ip/attunityreplicate/5.5.118/#

Enable Different Console Configurations in a High Availability Environment

In a High Availability active-passive scenario, it is recommended to install the Attunity Replicate data folder on a shared storage device. The data folder contains various configuration files in XML format (e.g. `ServiceConfiguration.xml`). In a standard installation, you do not need to change the names of these files. However, in a High Availability environment in which Attunity Replicate Console is installed on two different machines, you may want each machine to have its own unique settings. In such a situation, changing the name of these files to include the hostname of the Attunity Replicate Console machine (as it appears in the output of the Windows hostname command) will allow you to store a different configuration for each machine.

The file name should be formatted as follows: `[Name}Configuration-[hostname].xml`

For example, let's assume that one Attunity Replicate Console is installed on a machine called `replicate-main` and the other Attunity Replicate Console is installed on a machine called `replicate-failover`. To set up a different configuration for each machine, simply create two copies of the `ServiceConfiguration.xml` file and then rename them to `ServiceConfiguration-replicate-main.xml` and `ServiceConfiguration-replicate-failover.xml` accordingly. After renaming the files, edit each of them with your desired settings.

Setting up Attunity Replicate in a Linux Cluster

This section describes how to set up Attunity Replicate in a Linux Cluster Environment. There are several commercially available clustering solutions for Linux including Veritas Cluster Server, Red Hat Cluster Suite and IBM HACMP for Linux.

When one of the available clustering solutions is already in place, Attunity Replicate can be set up like any other cluster application while adhering to the following guidelines:

- Replicate should be installed on the participating cluster nodes.

  For more information, see Attunity Replicate on Linux: Installing, Upgrading and Uninstalling.
- Replicate only supports the failover cluster configuration (active-passive).
- Attunity Replicate data (the data folder, tasks folder, file channel, etc.) should be stored in a SAN for shared access between the cluster nodes. To change the default location of the Attunity Replicate data folder, run the following command on the primary cluster node when the installation completes:

  ```
  ./repctl-d <shared_storage_path> service start
  ```
- Only one instance of Replicate can be active at a given data location. The cluster software should be set so that during failover, one Replicate instance is stopped and the other is started.

F | Control Tables

This section describes the Attunity Replicate Control Tables that are created on the target endpoint when the corresponding table is selected in the **Control Tables** tab.

Note All Control Table timestamps are in UTC format.

In this appendix:

Apply Exceptions

Change Processing errors are recorded in the `attrep_apply_exceptions` table, which is described below.

The data in this table is never deleted.

Table F.1 | attrep_apply_exceptions Table

Column	Type	Description
TASK_NAME	nvchar	The name of the Attunity Replicate task.
TABLE_ OWNER	nvchar	The table owner.
TABLE_NAME	nvchar	The table name.
ERROR_TIME	timestamp	The time the exception (error) occurred.
STATEMENT	nvchar	The statement that was being executed when the error occurred.
ERROR	nvchar	The actual error.

Replication Status

The `attrep_status` table contains the current status of each replication task and the target data. Although updates for the tasks in the `attrep_status` table are generated every few seconds, Replicate will only apply the updates *after* it has applied changes to the target tables. In some cases, this may take a few minutes.

Table F.2 | attrep_status Table

Column	Type	Description
SERVER_NAME	nvchar	The name of the machine on which Attunity Replicate is installed.
TASK_NAME	nvchar	The name of the Attunity Replicate task.
TASK_STATUS	varchar	One of the following: » FULL LOAD » CHANGE PROCESSING Task status is FULL LOAD as long as there is at least one table in full load. After *all* tables have been loaded, the task status changes to CHANGE PROCESSING.
STATUS_TIME	timestamp	When the status was last updated.
PENDING_CHANGES	int	The number of change records that were not yet applied to the target.
DISK_SWAP_SIZE	int	The amount of disk space that is occupied by old or offloaded transactions.
TASK_MEMORY	int	Current memory consumption in MB.
SOURCE_CURRENT_POSITION	varchar	The POSITION in the source endpoint that Attunity Replicate is currently reading from.
SOURCE_CURRENT_TIMESTAMP	timestamp	The TIMESTAMP in the source from which Attunity Replicate is currently reading.
SOURCE_TAIL_POSITION	varchar	The POSITION of the oldest start transaction that is still not committed. This represents the newest position that you can revert to, without losing any changes. There may, of course, be duplicates.
SOURCE_TAIL_TIMESTAMP	timestamp	The TIMESTAMP of the oldest start transaction that is still not committed. This represents the newest TIMESTAMP that you can revert to, without losing any changes. There may, of

Table F.2 | attrep_status Table (Cont.)

Column	Type	Description
		course, be duplicates.
SOURCE_ TIMESTAMP_ APPLIED	timestamp	This is the timestamp of the last transaction committed. In a bulk apply this will be the timestamp for the commit of the last transaction in that batch. It will only be changed as part of the last transaction in the batch.

Suspended Tables

When a table is suspended, information about the table including the reason for its suspension is recorded in the `attrep_suspended_tables` table. If a table is suspended and then unsuspended while the task is running, the entries for that table will be deleted from the `attrep_suspended_tables` table.

When a task with suspended tables stops, records for that task will remain in the `attrep_suspended_tables` table. In the event that the tables are still suspended when the task is restarted, the data is deleted and then recorded again.

Table F.3 | attrep_suspended_tables Table

Column	Type	Description
SERVER_NAME	nvchar	The name of the machine on which Attunity Replicate is installed.
TASK_NAME	nvchar	The name of the Attunity Replicate task.
TABLE_OWNER	nvchar	The owner of the suspended table.
TABLE_NAME	nvchar	The name of the suspended table.
SUSPEND_REASON	varchar	The reason why the table was suspended.
SUSPEND_ TIMESTAMP	timestamp	The date and time the table was suspended.

Replication History

The `attrep_history` table provides statistics about each task, such as the number and volume of records processed during a particular timeslot.

A new record is appended to the table at the end of each TIMESLOT_DURATION. In other words, the data in this table is never deleted.

Table F.4 | attrep_history Table

Column	Type	Description
SERVER_NAME	nvchar	The name of the machine on which Attunity Replicate is installed.
TASK_NAME	nvchar	The name of the Attunity Replicate task.
TIMESLOT_TYPE	varchar	One of the following: » FULL LOAD » CHANGE PROCESSING (CDC) When FULL LOAD and CHANGE PROCESSING are running in parallel (some tables in full load, some in CDC), two history records will occupy the same time slot.
TIMESLOT	timestamp	The end timestamp of the time slot.
TIMESLOT_DURATION	int	The duration of each history record in minutes.
TIMESLOT_LATENCY	int	The latency at the end of the time slot. This is only applicable to CDC time slots. Note that this value contains the value of the *target* latency only.
RECORDS	int	The number of records processed during the time slot.
TIMESLOT_VOLUME	int	The volume of data processed in MB.

Change Data Partitions

The **attrep_cdc_partitions** table contains records of partitions created on the target database when Change Data Partitioning is enabled for a Replicate task. You can use this information to identify partitioned data that needs to be further processed.

Table F.5 | attrep_cdc_partitions Table

Column	Type	Description
SERVER_NAME	STRING	The name of the machine on which Attunity Replicate is installed.
TASK_NAME	STRING	The name of the Attunity Replicate task.
PARTITION_NAME	STRING	The partition name consists of the partition start and end time. **Example:**

Table F.5 | attrep_cdc_partitions Table (Cont.)

Column	Type	Description
		20170313T123000_20170313T170000
PARTITION_START_TIME	TIMESTAMP	When the partition was opened: **Example:** 2017-03-13 12:30:00.000
PARTITION_END_TIME	TIMESTAMP	When the partition was closed: **Example:** 2017-03-13 17:00:00.000
TABLE_OWNER	STRING	The table schema or owner.
TABLE_NAME	STRING	The table name.

DDL History

The **attrep_ddl_history** table contains a history of DDL changes that occurred in the source during replication to the target.

Note Currently, the DDL History table is only supported with the Hadoop target endpoint.

A new record is inserted into the table whenever a supported DDL change occurs in the source. Multiple ALTER TABLE statements that occur during a task may be represented as a single row in the control table. The JSON buffer (see below) describes all the changes that occurred (e.g. ADD COLUMN A, DROP COLUMN B, ALTER COLUMN C).

For information on enabling the DDL History Control Table as well as its limitations, see Control Tables.

Note When the **Apply Changes** task option is enabled, an entry is created for the base table (e.g. tblT1). If the **Store Changes** task option is also enabled, an additional entry is created for the CT table (e.g. tblT1__CT).

Table F.6 | attrep_ddl_history

Column	Type	Description
SERVER_NAME	STRING	The name of the machine on which Attunity Replicate is installed.
TASK_NAME	STRING	The name of the Attunity Replicate task.

Table F.6 | attrep_ddl_history (Cont.)

Column	Type	Description
TABLE_OWNER	STRING	The source table schema or owner.
TABLE_NAME	STRING	The source table name. If the table was renamed, this will be the table name before the change.
CHANGE_SEQ	STRING	See Change_Seq in Using Change Tables.
TIMESTAMP	TIMESTAMP	When the change occurred.
TABLE_VERSION	INTEGER	Replicate assigns an internal version number to the table. The version number increases whenever a DDL change occurs in the source table.
DDL_TYPE	STRING	CREATE_TABLE, DROP_TABLE, ALTER_TABLE, TRUNCATE_TABLE

DETAILS	CLOB	JSON document describing the change(s)
		Example:
		`{`
		`"owner": "string",`
		`"table": "string",`
		`"tableNewName": "string",`
		`"version": number,`
		`"columns":`
		`{`
		`"col1":`
		`{`
		`"columnNewName": "string",`
		`"action": "string",`
		`"type":" string",`
		`"length": number,`
		`"precision": number,`
		`"scale": number,`
		`"primaryKeyPosition": number,`
		`"nullable": boolean,`
		`"ordinal": number`
		`},`
		`"col2":`
		`{`
		`…`
		`}`
		`}`
		`}`

Table F.6 | attrep_ddl_history (Cont.)

Column	Type	Description
		where: » `tableNewName` is NULL if the table was not renamed » `col1` and `col2` are the original column names (or the *only* column names if the columns were not renamed) » `columnNewName` is the new column name or NULL if the column was not renamed » `action` is ADD (also for CREATE TABLE), DROP, or ALTER » `type` is the Replicate data type » `primaryKeyPosition` is ZERO if the column is not part of the primary key

G | Using HP NonStop SQL/MP as an ODBC Target

HP NonStop SQL/MP can be used as an ODBC target in a Replicate task. However, to ensure that Replicate and non-HP NonStop SQL/MP sources comply with HP NonStop SQL/MP conventions, several additional steps need to be performed. In addition, when defining the task settings, certain options will be unavailable or limited in their availability.

Note that when replicating to an HP NonStop SQL/MP target, you must select **SQLMP (ARC)** as the provider type. For information on how to select a provider type as well as a detailed explanation of how to define HP NonStop SQL/MP as an ODBC target in a Replicate task, see Using ODBC to Connect to a Target.

In this appendix:

Prerequisites

Before designing a task with HP NonStop SQL/MP as an ODBC Target, the following prerequisites must be met:

- All source tables and their indexes must already exist on the target database.
- The default name for the Replicate Apply Exceptions table (`attrep_apply_exceptions`) is not compliant with HP NonStop SQL/MP table naming conventions which dictate that table names cannot exceed 8 characters or contain non-alphanumeric characters. To resolve this conflict, you need to add the following entry to your NAVMAP file or create a new NAVMAP file (The NAVMAP file is located in the subvolume where ARC is installed):

  Syntax:

  ```
  [SQLMP]
  ```

  attrep_apply_exceptions= \machine_name.$volume_name.subvolume_name.atreapex

  You also need to manually create the Apply Exceptions table on the HP NonStop SQL/MP target *before starting the task*. The table should be named `atreapex` and should contain the following columns:

Column Name	Data Type
TASK_NAME	Varchar(128)
TABLE_OWNER	Varchar(128)
TABLE_NAME	Varchar(128)
ERRORTIME	Datetime year to fraction
STATEMENT	Varchar(1000)
ERROR	Varchar(1000)

- If you intend to enable the Replicate Store Changes option, you need to manually create the audit table on the HP NonStop SQL/MP target *before starting the task*.

Note This also requires you make the following changes in the Store Changes Settings tab:

- Select **Audit table** from the **Store changes** drop-down list.
- Change the default audit table name from `attrep_audit_table` to `atreauta`.

Create a table named `atreauta` with the following parameters:

Column Name	Data Type
*task_name	Varchar(110)
*stream_position	Varchar(128)
change_seq	Varchar(35)
change_oper	Varchar(1)
schema_name	Varchar(128)
table_name	Varchar(128)
operation	Varchar(12)
transaction_id	Varchar(32)
timestamp	Timestamp
change_record	Varchar(1000)
bu_change_record	Varchar(1000)

Note : You also need to create a Unique Index consisting of the **task_name** and **stream_position** columns (marked with an asterisk above).

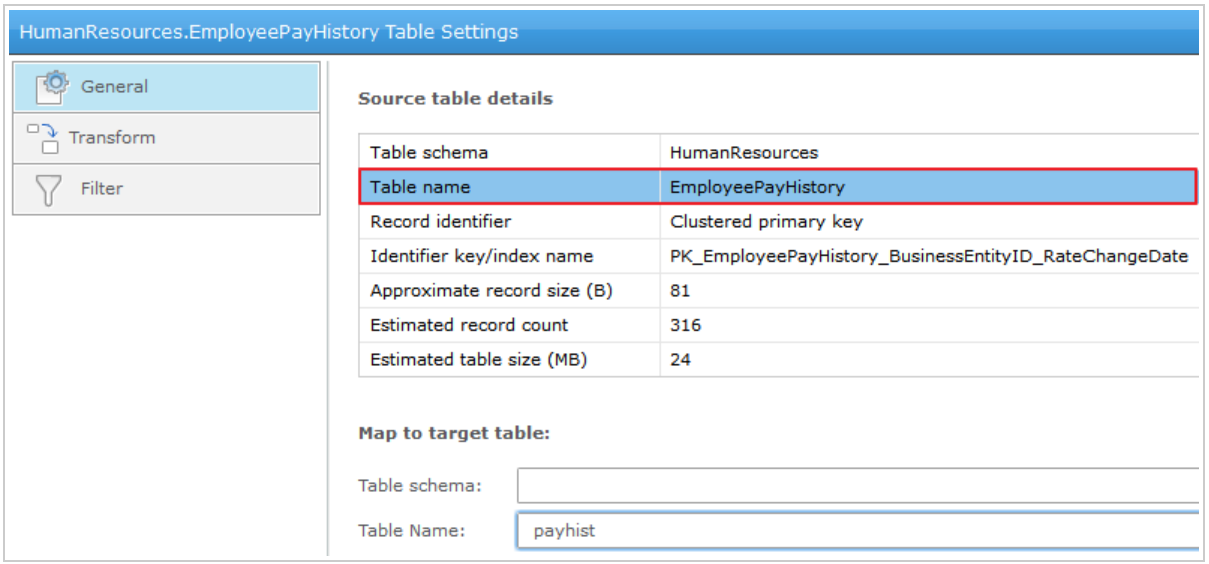
Table Settings

When the source database in a Replicate task is not HP NonStop SQL/MP, you must make sure that the selected source tables comply with HP NonStop SQL/MP conventions and limitations. This section describes how to apply transformations to source tables that are not HP NonStop SQL/MP-compliant.

- The maximum size of a Unique Index in HP NonStop SQL/MP cannot exceed 240 bytes. In the Transform tab of the **<Table_Name> Table Settings** dialog box, you can check which columns comprise the Unique Index and change them if necessary (i.e. if their combined **Type** value exceeds 240 bytes). For an explanation of how to do this, see Using the Transform Tab.
- HP NonStop SQL/MP does not support UPDATE operations on Primary Key columns. Therefore, if your application updates source tables columns that are part of a Primary Key, you will need to create the target table with a Unique Index on those columns instead of a Primary Key. For an explanation of how to do this, see Using the Transform Tab.
- Valid HP NonStop SQL/MP table names cannot exceed 8 characters or contain non-alphanumeric characters. If any of the source table names are not HP NonStop SQL/MP-compliant, you will need to map them to a valid name.

  To do this:

  - Open the **<Table_Name> Table Settings** dialog box as described in Table Settings.

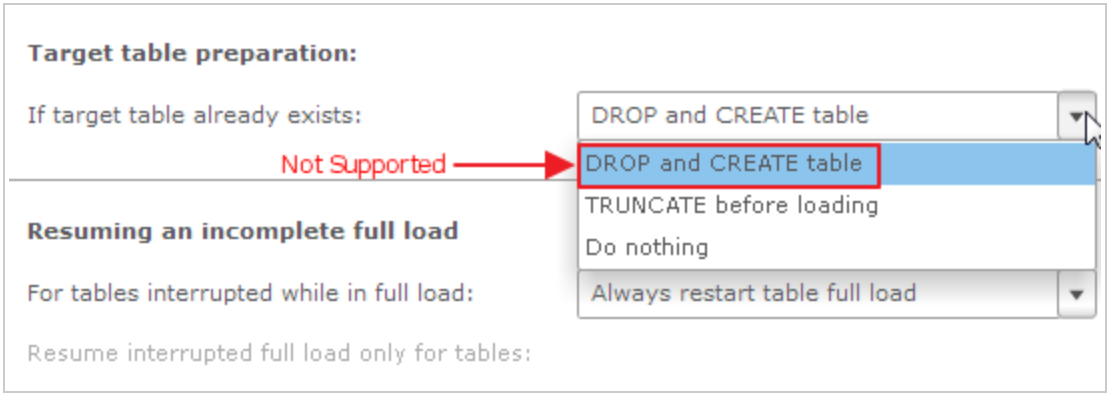


  - In the **General** tab's **Table Name** field, enter the new table name (as shown above).

Task Setting Limitations

When defining your task settings, the following limitations apply:

- **Metadata:**
  - **Target Metadata:** As HP NonStop SQL/MP does not support LOB data types, the setting in this tab are not relevant.
  - **Control Tables:** None of the optional tables are supported. The Apply Exceptions table, which is required, should be configured as described in Prerequisites above.
- **Full Load:**
  - **Full Load Settings:** As the source tables need to be created manually on the target, the **DROP and CREATE table** option (shown below) is not supported.

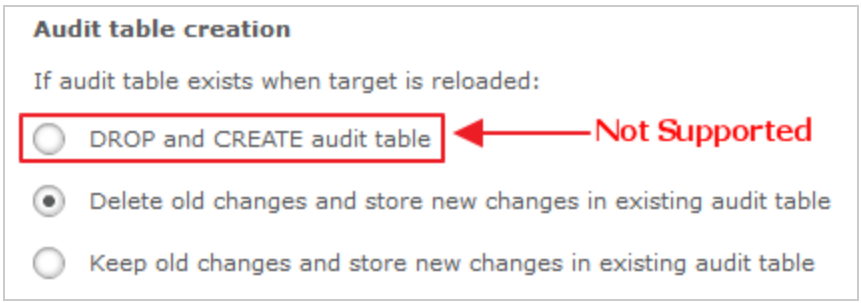


  - **Full Load Tuning:** No limitations.
- **Change Processing:**
  - **Apply Changes Settings:** No limitations
  - **Store Changes Settings:** Change tables are not supported. If you want Replicate to store captured changes on HP NonStop SQL/MP, choose **Audit table** from the **Store changes in** drop-down list. This also requires you to manually create the Audit Table on HP NonStop SQL/MP *before starting the task*, as described in Prerequisites above.

    After creating the Audit table, specify its name in the **Audit table name** field (instead of the default name).

    Audit table creation options:

    As the Audit Table is created manually, the **DROP and CREATE audit table** option (shown below) is not supported.



- **Change Processing Tuning:** Only "Transactional apply" **Change processing mode** is supported.

- **Error Handling:** No limitations.
- **Logging:** No limitations.

For a detailed description of Task Settings, see Task Settings.

H | Impact of DST Change on Attunity Replicate

This section describes how Attunity Replicate is affected by Daylight Saving Time (DST) and provides guidelines for handling changes brought about by DST.

There are two types of DST changes:

- **DST On** - Occurs approximately when Summer starts (actual date is country specific). Its impact on local time is that local time is moved one hour forward (so, for example, 01:00 AM becomes 02:00 AM). This DST change does not impact Attunity Replicate because it does not result in time overlap.
- **DST Off** - Occurs approximately when Winter starts (actual date is country specific). Its impact on local time is that local time is moved back one hour (so, for example, 02:00 AM becomes 01:00 AM). This DST change results in time overlap where local time travels over the same hour twice in a row.

The comments below assume that the customer has not changed the time but rather the timezone or the DST setting. Changing the actual time (not for minor time adjustments) is a sensitive operation and is best done when Attunity Replicate is stopped.

Running Attunity Replicate tasks do not depend on the timezone or DST for correctly scanning and processing the transaction logs. Internally, Attunity Replicate timers use UTC.

Still, there are several places where DST may have an effect:

1. Timestamps in logs and audit messages are in local time. As a result, when Winter time starts, the logs will show the time going back an hour; conversely, when Summer time starts, the logs may appear to be missing one hour.
2. Scheduled jobs as well as the global and table manipulation variables `timestamp` and `commit_timestamp` use local time so these will also be affected. The impact of this depends on the manipulation done and on the intended use of the timestamp based data.

Important: To prevent timestamp and scheduling anomalies resulting from DST starting or ending, the following best practices should be observed:

- DST Off (summer to winter): Do not schedule a task to start from the time the clock changes until the following hour. For example, if DST ends at 02:00 am, do not schedule a task to run between 02:00 and 02:59, as the task will run twice.
- DST On (winter to summer): Do not schedule a task to start from the time the clock changes until the following hour. For example, if DST starts at 02:00 am, do not schedule a task to run between 02:00 and 02:59 as this hour does not exist.

If you have existing jobs scheduled to start at the overlap time and you do not want to modify them, then you need to stop the Attunity Replicate Server. Going in to

Winter time, for example, if at 02:00 AM the clock is to be set back to 01:00 AM then when the time is 00:55 AM the Attunity Replicate Server should be stopped and, after an hour and ten minutes (at 01:05 AM), should be started again.

If you forget to do this, all scheduled jobs will run an hour earlier than intended. You can rectify this by setting the desired scheduling time and then restarting the Attunity Replicate Server service.

3. Statistics shown on the console are also sensitive to local time and thus may also show confusing/inaccurate data in the overlap period (going in to Winter time) or for the skipped period (going into Summer time).
4. If the clock on Attunity Replicate Server machine is one hour behind the clock on the Attunity Replicate Console (UI) machine, the following issues are known to occur:
   - The Applied Changes circle graph will be updated as the changes are applied, but the information in the Recent Activity tab will not be updated.
   - Scheduled jobs will start according to the Attunity Replicate Server time (as expected), but will remain in the **Active Jobs** list after execution instead of moving to the **Expired Jobs** tab.

     For more information on scheduling jobs, see Scheduling Jobs.

In general, it is recommended to avoid non-critical task design changes during the first overlap period (going in to Winter time) so as to prevent confusion about when the changes took place.

In addition to Attunity Replicate, other components are also involved including:

- The source endpoint system
- The target endpoint system
- The local operating system
- The task design (specifically using timestamp based variables)

Given the complexity of the topic and the involvement of many independent components and settings, Attunity generally recommends that customers first verify the impact of DST changes in their test environment.

I | Metadata File Description

When the Create metadata files in the target folder option in the File target endpoint or Amazon S3 target endpoint is selected, for each CSV/JSON file Replicate creates a corresponding metadata file under the specified target folder.

The metadata file offers several benefits such as enabling custom batch processes to perform better validation, supporting deeper automation, offering lineage information and improving processing reliability.

The metadata files (which are in standard JSON format) are described in the table below.

Note All timestamps are in ISO-8601 format, for example `2016-08-02T10:05:04.802`.

Field	Description
Task Information	
name	The name of the Replicate task.
sourceEndpoint	The name defined in the source endpoint settings.
sourceEndpointType	The endpoint type defined in the source endpoint settings (e.g. Oracle, MySQL, etc.).
sourceEndpointUser	The user defined in the source endpoint settings.
replicationServer	The hostname of the machine on which Attunity Replicate is installed.
operation	If a target data file has been created, this field will contain the following value: `dataProduced`
File Information	
name	The name of the data file without the extension.
extension	The extension of the data file (.csv or.json according to the selected target file format).
location	The location of the data file.
startWriteTimestamp	UTC timestamp indicating when writing to the file started.
endWriteTimestamp	UTC timestamp indicating when writing to the file ended.
firstTrans-actionTimestamp	UTC timestamp of the first record in the file.
lastTrans-actionTimestamp	UTC timestamp of the last record in the file.

Field	Description
content	The values can either be `data` (i.e.Full Load replication) or `changes` (i.e Change Processing replication) according to the data in the corresponding CSV file.
recordCount	The number of records in the file.
errorCount	The number of data errors encountered during file creation.
Format Information	
format	`delimited` or `json` according to the selected target file format.
options	The options for `delimited` file format. These options will not be shown for `json` format as they are not relevant.
recordDelimiter	The delimiter used to separate records (rows) in the target files. The default is newline (`\n`).
fieldDelimiter	The delimiter used to separate fields (columns) in the target files. The default is a comma.
nullValue	The string used to indicate a null value in the target file.
quoteChar	The character used at the beginning and end of a column. The default is the double-quote character (").
escapeChar	The character used to escape a string when both the string and the column containing the string are enclosed in double quotes. Note that the string's quotation marks will be removed unless they are escaped. **Example (where " is the quote character and \ is the escape character):** 1955,"old, \"rare\", Chevrolet",$1000
Custom Information	
customInfo	This section contains any custom parameters that were set using the **dfmCustomProperties** internal parameter. The **dfmCustomProperties** internal parameter must be specified in the following format: `Para-`

Field	Description

	`meter1=Value1;Parameter2=Value2;Parameter3=Value3`
	
	Example:
	`Color=Blue;Size=Large;Season=Spring`
	
	For an explanation of how to set internal parameters, see Internal Parameters.

Data Information	
sourceSchema	The schema containing the source table.
sourceTable	The name of the source table.
targetSchema	The name of the target table schema (if the source schema name was changed). For information on changing the source schema name, see Carrying out General Tasks for a Single Table/View.
targetTable	The name of the target table (if the source table name was changed). For information on changing the source table name, see Carrying out General Tasks for a Single Table/View.
tableVersion	Replicate assigns an internal version number to the table. The version number increases whenever a DDL change occurs in the source table.
columns	Information about the table columns.
ordinal	The position of the column in the record (1, 2, 3, etc.).
name	The column name.
type	The column data type. See File Target Data Types or Amazon S3 Target Data Types for more information.
width	The maximum size of the data (in bytes) permitted for the column.
scale	The maximum number of digits to the right of the decimal point permitted for a number.
primaryKeyPos	The position of the column in the table's Primary Key or Unique Index. The value is zero if the column is not part of the table's Primary Key.

J | Supported Platforms and Endpoints

In addition to listing the platforms on which Attunity Replicate can be installed, this appendix specifies which source and target endpoint versions can be used in an Attunity Replicate task.

In this appendix:

Supported Windows Platforms

Attunity Replicate can be installed on any of the following Windows platforms:

- Windows 7 (64-bit)
- Windows Server 2008 R2 (64-bit)
- Windows Server 2012 (64-bit)
- Windows Server 2012 R2 (64-bit)
- Windows Server 2016 (64-bit)

Supported Linux Platforms

Attunity Replicate can be installed on the following Linux platforms:

- Red Hat Enterprise Linux 6.2, 6.3, 6.4, 6.5, 6.6, 6.7, 6.8, 7.0, 7.1, 7.2, and 7.3 (64-bit)
- SUSE Linux 11.4 or 12.0 (64-bit)

Supported Source Endpoints

The table below lists the source endpoint versions supported by Attunity Replicate.

Endpoint	Version	Windows	Red Hat Linux	SUSE Linux
Cloud-Based				
Amazon RDS for Aurora	Compatible with MySQL 5.6	✓	✓	✓
Amazon RDS for MySQL	5.6 and 5.7	✓	✓	✓
Amazon RDS for Oracle	11.2.0.2.v7	✓	✓	✓
ARC-Based				
IBM DB2 for iSeries (ARC)	**OS/400:** 5.8, 6.1, 7.1 and 7.2	✓	✓	✓
IBM DB2 for z/OS (ARC)	**IBM z/OS:** 1.12, 1.13 and2.1 **IBM DB2:** 9, 10 and 11	✓	✓	✓
HP Nonstop SQL/MP (AIS) HP Nonstop Enscribe (AIS)	**Himalaya:** G06.32 **Itanium:** H06.22/J06.14	✓	✓	✓
OpenVMS RMS (ARC)	**Alpha:** 8.3 **Itanium:** 8.3	✓	✓	✓
IBM IMS (ARC)	**IBM z/OS:** 1.12, 1.13 and 2 **IMS:** 11, 12 and 13	✓	✓	✓
IBM VSAM Batch (ARC)	**IBM z/OS:** 1.12, 1.13 and 2.1	✓	✓	✓
File				

Endpoint	Version	Windows	Red Hat Linux	SUSE Linux
File	N/A	✓	✓	✓
File Channel	N/A	✓	✓	✓
Hadoop				
Hadoop Cloudera	5.5, 5.6, 5.7, 5.8, 5.9, 5.10, and 5.11	✓	✓	✓
Hadoop Hortonworks	2.2, 2.3, 2.4, 2.5.x, 2.6	✓	✓	✓
Hadoop MapR	4.0, 4.1, 5.0, 5.1, and 5.2	✓	✓	✓
Relational Databases				
IBM DB2 for LUW	9.7 Fix Pack 1, 10.1, 10.5 **Note** 10.5 with fix pack 5 is *not* supported.	✓	✓	✓
IBM DB2 for z/OS	10 and 11	✓	✓	✓
IBM DB2 for iSeries	7.1, 7.2 and 7.3	✓	✓	✓
IBM Informix	11.5, 11.70, 12, 12.10	✓		
Microsoft SQL Server	2005, 2008, 2008 R2, 2012, 2014, 2016	✓	✓	✓
MySQL	5.5, 5.6, and 5.7	✓	✓	✓
MariaDB	10.0.24 to 10.0.28	✓		
Percona	5.6.28		✓	✓
Oracle	10.x, 11.x, 12.1 and 12.2* *Using Replicate to move data from an Oracle 12.2 database is currently in beta status.	✓	✓	✓
PostgreSQL	9.4.2 and 9.4.5, 9.5, and 9.6	✓	✓	✓
SAP Sybase ASE	12.5, 15, 15.5, 15.7, 16	✓		

Endpoint	Version	Windows	Red Hat Linux	SUSE Linux
Data Warehouses				
Teradata Database	13, 14, 14.10, 15, 15.10	✓	✓	✓
ODBC				
ODBC	3.0, 3.5	✓	✓	✓
ODBC with CDC	3.0, 3.5	✓	✓	✓
Other				
Salesforce	N/A	✓		
SAP Application	Supported backend databases: Oracle, Microsoft SQL Server, and IBM DB2 for LUW. See Relational Databases above for version and platform information.			

Supported Target Endpoints

The table below lists the target endpoint versions supported by Attunity Replicate.

Endpoint	Version	Windows	Red Hat Linux	SUSE Linux
Cloud-Based				
Amazon RDS for Aurora	Compatible with MySQL 5.6	✓	✓	✓
Amazon RDS for MariaDB	N/A	✓	✓	✓
Amazon RDS for Microsoft SQL Server	2008 R2 and 2012	✓		
Amazon RDS for MySQL	5.6 and 5.7	✓	✓	✓
Amazon RDS for Oracle	11.2.0.2.v7	✓	✓	✓
Amazon RDS for PostgreSQL	9.4.1	✓	✓	✓
Amazon Redshift	N/A	✓	✓	✓
Snowflake	N/A	✓	✓	✓
Amazon S3 **Note** This endpoint is currently in Beta status.	N/A	✓	✓	✓
Google Cloud SQL	N/A	✓	✓	✓
Microsoft Azure SQL Data Warehouse	N/A	✓		
Microsoft Azure SQL Database	Supported via the Microsoft SQL Server endpoint.	✓		
Microsoft Azure Event Hubs **Note** This endpoint is currently in Preview status.	N/A	✓		
File-Based				
File	N/A	✓	✓	✓
File Channel	N/A	✓	✓	✓

Endpoint	Version	Windows	Red Hat Linux	SUSE Linux
Hadoop				
Cloudera	5.5, 5.6, 5.7, 5.8, 5.9, 5.10, and 5.11	✓	✓	✓
Hortonworks	2.2, 2.3, 2.4, 2.5.x, 2.6	✓	✓	✓
MapR	4.0, 4.1, 5.0, 5.1, and 5.2	✓	✓	✓
Amazon EMR	5.2.0	✓	✓	✓
Data Warehouses				
Actian Vector	3.0, 3.5	✓	✓	✓
HP Vertica	6, 7, and 8.0.	✓	✓	✓
IBM Netezza	6.x and 7.x	✓		
Microsoft APS PDW	AU2	✓		
Pivotal Greenplum	4.2, 4.3	✓	✓	✓
Pivotal HAWQ	1.2.1	✓	✓	✓
SAP Sybase IQ	15.x	✓		
Teradata Database	13, 14, 14.10, 15, 15.10	✓	✓	✓
Teradata Aster	5.0	✓		
Relational Databases				
Microsoft SQL Server	2005, 2008, 2008R2, 2012, 2014, 2016	✓	✓	✓
MySQL	5.5 and 5.6	✓	✓	✓
MariaDB	10.0.24 to 10.0.28	✓		
Oracle	10.x, 11.x, 12.x	✓	✓	✓
PostgreSQL	9.0, 9.3.x, 9.4.x, 9.5.x, and 9.6	✓	✓	✓
SAP Sybase ASE	15, 15.5, 15.7, 16	✓		
SAP HANA	2.0	✓	✓	✓

Endpoint	Version	Windows	Red Hat Linux	SUSE Linux
Note This endpoint is currently in Preview status.				
ODBC				
ODBC	3.0 and 3.5	✓	✓	✓
NoSQL				
MongoDB	2.6 and 3.0	✓		
Kafka	0.8, 0.9 and 0.10	✓	✓	✓

Endpoints Supported in Bidirectional Replication

Bidirectional tasks support the following endpoints:

Source Endpoints:

- Oracle
- Microsoft SQL Server
- MySQL
- PostgreSQL
- All AIS sources
- File Channel
- SAP Sybase ASE

Target Endpoints:

- Oracle
- Microsoft SQL Server
- MySQL
- PostgreSQL
- ODBC
- File Channel
- SAP Sybase ASE

Supported Browsers

The following browsers are supported:

- Microsoft Internet Explorer version 11 or above
- Mozilla Firefox
- Google Chrome

Note Displaying the console in a window that spans multiple vertical windows is not supported.

The Attunity Preview Program

The Attunity Preview program is designed to provide early access to new functionality or integrations with third-party technologies (e.g. new endpoints). Selected customers will work closely with Attunity product management and provide feedback as the new capabilities progress through our development lifecycle.

K | Best Practices when Working with Oracle ASM

This appendix provides detailed guidelines for configuring Attunity Replicate to work with the Oracle source endpoint when the redo logs are stored in Oracle ASM.

When the redo logs are stored in Oracle ASM, it is recommended to use the **Copy redo logs to temporary folder** option available in the **Advanced** tab. This option is only available when the **Binary reader** access method is selected.

Binary Reader is Attunity's proprietary redo log parser for parsing the Oracle redo logs. Binary Reader provides greatly improved performance and reduces the load on the Oracle server when compared with other methods such as LogMiner.

In this appendix:

The "Copy redo logs to temporary folder" Method

When the **Copy redo logs to temporary folder** option is enabled, Replicate instructs Oracle to copy the full redo log or chunks of the redo log to a local folder residing on the Oracle Server machine or to a shared network folder that can be accessed by Oracle Server.

Chunks of redo logs are copied from the ASM online redo logs using the Oracle **DBMS_DISKGROUP** package whereas archived redo logs are copied in their entirety using the **COPY_FILE** method of the **Oracle DBMS_TRANSFER** package. In addition, Replicate uses Oracle directory objects to denote the source and target folder in this transfer (ASM is the source and the temporary folder is the target)

After the redo logs have been copied to the temporary folder, Replicate reads them using one of the following methods:

- **Method 1:** Using Oracle BFILE to read from files or partial files from Oracle directory objects.

- **Method 2:** Directly from the temporary folder after providing Replicate with access to the temporary folder. Using this method, Replicate reads the physical files directly from the folder.

Both options are significantly faster than accessing the redo logs directly from Oracle ASM. However, the "Direct Access" option using a shared network folder is the fastest and has the least impact on Oracle Server resources. This is because Replicate reads the files directly from the shared folder, thereby eliminating the need for Oracle to send Replicate the file content. However, using the "Direct Access" option requires some additional configuration (as described below).

To prevent old redo logs from accumulating in the temporary folder, Replicate should be configured to delete the redo log files from the temporary folder once they have been processed. The delete operation is performed using Oracle file groups and the Oracle DBMS_FILE_GROUP package.

Oracle Permissions Required for the Binary Reader and the "Copy redo logs to temporary folder" Options

Certain Oracle permissions are required, regardless of which access method is selected. These permissions are fully documented in Using Oracle as a Target.

This section describes only those permissions required for the Binary Reader and Copy redo logs to temporary folder options.

The following permissions are required:

- CREATE SESSION
- SELECT ON v_$transportable_platform
- EXECUTE ON DBMS_FILE_TRANSFER - Enables the archived redo logs to be copied to the temporary folder (using the COPY_FILE method)
- As mentioned above, Replicate needs to use directory objects for copying redo logs to the temporary folder and deleting them from the temporary folder. If you want Replicate to create and manage the Oracle directories, you need to grant the CREATE ANY DIRECTORY privilege. Replicate will then create the Oracle directories with the "attrep_" prefix. If you do not grant this privilege, you will need to create the corresponding directories manually. The directories should point to the archived redo logs and temporary folder paths. Do not append the 'attrep_' prefix to the directory names as Replicate ignores such object directories.
- If you create the directories manually and the Oracle user specified in the Oracle Source endpoint is not the user that created the Oracle directories, grant the READ ON DIRECTORY privilege as well.
- If the Oracle user specified in the Oracle source endpoint is not the user that created the Oracle directories, the following additional permissions are required:
  - READ on the Oracle directory object specified as the source directory (i.e. the ASM archived redo logs path and the temporary folder in the event that the BFILE method is used to read from the temporary folder)

- WRITE on the directory object specified as the destination directory in the copy process (i.e. the temporary folder).

Permissions for Deleting the Processed Redo logs from the Temporary Folder

The following permissions are required:

- GRANT SELECT ON DBA_FILE_GROUPS

  Example:

  GRANT SELECT ON DBA_FILE_GROUPS to attu_user;

- GRANT EXECUTE on SYS.DBMS_FILE_GROUP

  Example:

  GRANT EXECUTE ON SYS.DBMS_FILE_GROUP to attu_user;

- EXECUTE DBMS_FILE_GROUP.GRANT_SYSTEM_PRIVILEGE with the system privilege 'MANAGE_ANY_FILE_GROUP' for the Replicate user.

  Example:

  execute DBMS_FILE_GROUP.GRANT_SYSTEM_PRIVILEGE (DBMS_FILE_GROUP.MANAGE_ANY_FILE_GROUP, 'attu_user', FALSE);

Oracle ASM Access Permissions

Replicate requires ASM access permissions in order to read the online redo logs from ASM (SYSASM or SYSADM privilege). This is because reading the online redo logs from ASM is performed using a function from the Oracle DBMS_DISKGROUP package, which requires the SYSASM or SYSADM privileges.

From Oracle 11g Release 2 (11.2.0.2), Attunity Replicate must be granted the SYSASM privilege in order to access the ASM account. For older supported versions, granting Attunity Replicate the SYSDBA privilege should be sufficient.

You can also validate ASM account access by opening a command prompt and issuing the following statements:

```
sqlplus asmuser/asmpassword@+asmserver as sysdba
```

-OR-

```
sqlplus asmuser/asmpassword@+asmserver as sysasm
```

Setting up the File Share if the "Direct Access" option was chosen

If you use the Replicate has file level access to temporary folder option, you will need to set up a shared network folder (using NFS or SAMBA for example) that can be accessed from all Oracle nodes.

The shared folder must allow write access and grant delete permission to the user and group on which the Oracle database is running. In addition, it must allow read access to Replicate Server. Make sure you are able to read the contents of the share when logging in to the Replicate Server machine with the user under which the Attunity Replicate Server service runs ('Attunity' user).

This configuration should be done by the customer's IT team, but Replicate Support is always available to assist.

Configuring the "Copy to Temp Folder" Option in Replicate

In the **Advanced** tab of the Oracle source endpoint:

1. Select **Binary reader** as the **Access redo logs via** method.
2. Provide the following information in the ASM parameters fields:
   - A connection string to ASM
   - A user and password for an Oracle user that has access to ASM (SYSASM or SYSADM - see privileges section above)
3. To access the temporary folder using BFILE, select the **Copy redo log files to temporary folder** check box and then specify the full path of the temporary folder path in the designated field (e.g. **/mnt/share**).
4. To access the temporary folder directly (using a shared network folder):
   a. Select the **Copy redo log files to temporary folder** check box and then specify the full path of the temporary folder path in the designated field (e.g. **/mnt/share**).
   b. Select the **Replicate has file level access to temporary folder** check box.
   c. Select the **Access Archived Redo logs in folder** check box and the specify the share pointing to the temporary folder in the designated field. For example, if the temporary folder **/mnt/share** is shared with the Replicate Linux machine as **/storage/ora_share**, enter **/storage/ora_share**.
5. Select the **Delete processed archived redo log files** check box.

Additional Considerations

This section describes additional factors that should be taken into consideration when working with Oracle ASM.

Security and Folder Location

Oracle database must be able to write to the temporary folder and delete from it.

If the Oracle source database is part of a RAC cluster, the temporary folder must be located on a file share that is accessible to all of the nodes in the Oracle Server RAC.

If you chose the "BFILE" method for accessing the temporary folder, the folder does not need to be a shared network folder. In such as case, only authorized Oracle users can access it and only through Oracle.

If you choose the "Direct Access" option, only the Oracle user and the "Remote" Replicate user (NFS or Simba user) that will read the redo log should be granted permission to access the folder.

Also, as the archived redo logs are deleted from the temporary folder after processing, the risk of unauthorized access to the redo logs is greatly diminished.

Multiple Tasks Using the Same Temporary Folder

If multiple tasks use the same temporary folder, conflicts may arise such as one task needing to access a redo log that another task has already deleted. To prevent such conflicts, only one Replicate task should access the same temporary folder at any given time. You can, however, create a different subfolder under the same root folder for each task. Then, for example, instead of specifying **/mnt/share** as the temporary folder for both tasks, you can specify **/mnt/share/task1** for one task, and **/mnt/share/task2** for the other.

Temporary Folder Disk Usage

When working with the temporary folder, you should enable the Delete processed archived redo log files option described in Configuring the "Copy to Temp Folder" Option in Replicate. This ensures that the redo logs will not accumulate in the temporary folder.

As Replicate needs to copy the redo logs from all your RAC nodes, you will need to allocate enough space in the temporary folder for (at least) the maximum size of your redo logs multiplied by the number of RAC nodes in your Oracle source. Also, for each temporary folder, allocate up to 50 MB multiplied by the number of RAC nodes. This is for small files which are copies of the chunk that Replicate is currently reading from the ASM online redo log. Although these files are not deleted, there will only be one such small file per RAC in each temporary folder at any given time.

In the event that multiple tasks are running, disk space calculation should also be multiplied by the number of tasks as each task has its own temporary folder.

Glossary

C

Change Data Capture (CDC)
Captures changes in the source data or metadata as they occur and applies them to the target endpoint as soon as possible, in near-real-time. The changes are captured and applied as units of single committed transactions and several different target tables may be updated as the result of a single source Commit. This guarantees transactional integrity in the target endpoint. The CDC process for any file or table starts as soon as the data Load operation for the file or table begins.

F

Full Load
Creates all defined files or tables at the target endpoint, automatically defines the metadata that is required at the target, and populates the tables with data from the source.

L

Latency
Latency can be understood as follows: - Source Latency: The gap in seconds between the original change in the source endpoint and capturing it. - Target Latency: The gap in seconds between the original change in the source endpoint and applying it to the target endpoint. - Apply Latency: The gap in seconds between capturing the change in the source endpoint and applying it to the target endpoint.

S

Source Endpoint
A collection of files or tables managed by an endpoint management system (such as, Oracle, SQL Server) that is part of the main computing service of the IT organization of an enterprise. This source continuously updated, may need to provide a high throughput rate, may have strict 24/7 up-time requirements, and may reference or update a number of tables in the course of a single logical transaction while providing transactional consistency and integrity for the data.

T

Target Endpoint
A collection of files or tables managed by an Endpoint Management System (DBMS), which may be different from the DBMS managing the source endpoint. It contains data that is derived from the source. It may contain only a subset of the tables, columns, or rows that appear in the source. Its tables may contain columns that do not appear in the source but are transformations or computations based on the source data.

Index

E

F

G

K

L

M

P

R

S

T

U

V

W