DIVA Connect

Installation, Configuration, and Operations Guide

Release 3.1.0

Version 1.1

January 2023

Copyrights and Trademark Notices

telestream | DIVA

Contents

3 Installing DIVA Connect

4 Configuring DIVA Connect Services

6 DIVA Connect Administration

7 DIVA API

8 Troubleshooting

A Sample Configuration Files

Glossary

List of Tables

Preface

This document describes how to install, configure, and administer DIVA Connect 3.1.0.

Notes: Refer to the DIVA Core Supported Environments book for minimum hardware and software requirements to run DIVA Connect.

Before DIVA Connect 2.0, DIVA Connect was installed, revised, and released with DIVA Core. DIVA Connect is now released and revised independently of DIVA Core. DIVA Connect 3.0.0 is not directly compatible with Legacy DIVA Connect 1.0 (in DIVA Core 6.x and 7.x), and is configured differently. It is therefore not a drop in replacement.

Documentation Accessibility

For information about Telestream's commitment to accessibility, visit the Telestream Support Portal located at https://www.telestream.net/telestream-support/diva/support.htm.

Access to Telestream Support

Telestream customers that have purchased support have access to electronic support through the Telestream Support Portal located at https://www.telestream.net/telestream-support/diva/support.htm.

Related Documents

For additional information, see the DIVA Connect Security Guide for this release for your installed DIVA Core release located at https://www.telestream.net/telestream-support/diva/support.htm.

Document Updates

The following table identifies updates made to this document.

Date	Update
February 2021	Rebranded document to Telestream Updated copyright notices
July 2021	Updated book for release 3.0.0
December 2021	Updated book for release 3.1.0
January 2023	Added note to refer to DIVA Core Supported Environments for minimum hardware and software requirements to run DIVA Connect.

Conventions

The following text conventions are used in this document:

Convention	Meaning
boldface	Boldface type indicates graphical user interface elements associated with an action, or terms defined in text or the glossary.
italic	Italic type indicates book titles, emphasis, or placeholder variables for which you supply particular values.
`monospace`	Monospace type indicates commands within a paragraph, URLs, code in examples, text that appears on the screen, or text that you enter.
blue text	Blue text indicates a link to an outside source, or to another chapter, section, or glossary term in this book.

1

Overview

System Overview

Note: Refer to the DIVA Core Supported Environments book for minimum hardware and software requirements to run DIVA Connect.

DIVA Connect provides a unified view of archived digital assets across multiple, distributed DIVA Core systems and the cloud. It facilitates the moving of content back and forth among DIVA Core sites, and from customer Server and disks. DIVA Connect assists with disaster recovery, content distribution, access control, performance, and content availability.

By connecting to multiple DIVA Core sites, DIVA Connect creates a virtual archive system that spans geographical locations. You can leverage disaster recovery sites to provide additional asset availability when outages occur, and improve archive and restore performance for some assets when systems run normally. You can use DIVA installations from other parts of your organization as strategic backup sites, content sharing partners, and participants in global business media workflows. DIVA Connect eases the burden of keeping track of content that could exist in local DIVA sites, remote DIVA sites, and the cloud.

Applications can use DIVA Core API to connect directly to DIVA Connect. This allows applications to store and access digital assets in different geographical locations. DIVA Connect synchronizes asset information from each DIVA site, so that it always has an up-to-date asset inventory. DIVA Connect uses this information to enforce naming uniqueness, and to select the best site for various requests (*for example Restore and Copy requests*). DIVA Connect also provides access rules to limit the operations that users are allowed to perform.

API Support

DIVA Connect allows applications written using the DIVA API to control multiple DIVA Core sites as if they were one large archive system. The DIVA Core API allows client applications to connect to an archive system and store content, copy content, delete content, or restore content (*see the DIVA Core C++ API Programmer's Guide for more on the DIVA Core API*). The

API also supports retrieving of information about archived objects. DIVA Connect implements support for a large subset of the DIVA API - that which is commonly used by popular applications such as MAM (*Media Asset Management*) applications.

DIVA Connect 3.1.0 supports the new REST API that is included as part of the installation of DIVA Core 8.0 and later. For more detailed implementation information, see the Swagger at https://[IP address of DIVA install]:8765/swagger-ui.html.

For more information, see Chapter 7.

User Interface (*DIVAConnectUI*)

DIVAConnectUI allows the user to monitor DIVA Connect requests, and view, copy, restore, and delete DIVA Connect Assets (*DIVA archived objects*) across multiple DIVA Core sites. All DIVA Connect level requests can be monitored, whether issued through the API or through the UI itself. You can also view asset information for all configured DIVA Core sites, whether the asset was archived through DIVA Connect or not. DIVAConnectUI provides flexible ways of querying both request information and asset information.

DIVA Connect 3.1.0 does not support the DIVA Core Control GUI (*even in Direct Mode*). Instead, use the DIVAConnectUI to monitor DIVA Connect, and the DIVA Core Control GUI to monitor DIVA Core.

For more information, see Chapter 5.

Command Line Interface (*DIVAConnectAdmin*)

You can use the DIVAConnectAdmin command-line tool to:

- Monitor the status of database synchronization
- Set site-specific parameters and limits
- View API connections
- Reload the workflow profiles and access rules
- Change the log level of DIVA Connect services

For more information, see Chapter 6.

Rule-Based Policies

Using DIVA Connect, you can create rules to control which users and which Workflow Profiles can perform certain operations. For instance, you can create rules that limit which Servers can receive restored content, or limit which objects can be copied to sites. For more information, see "Access Rules" on page 4-19.

You can use the AutoCopy feature to control which assets automatically replicate from one site to another. This feature supports flexible ways of scheduling copy tasks and can handle large numbers of requests. For more information, see "AutoCopy" on page 4-26.

Other DIVA Core Components

Some inter-site transfer tasks are not performed specifically by DIVA Connect, but instead are carried out by other DIVA Core components as noted below:

The DFM (*Drop Folder Monitor*) service monitors content that is being written to disk in a folder. It then archives that content into a DIVA Core system after the content has been completely transferred. DFM can be used as part of an intersite copy workflow, in DIVA

Connect. *See the DIVA Core DFM (Drop Folder Monitor) User's Guide in the DIVA Core Documentation library for more information on DFM*.

You can configure SPM (*Storage Plan Manager*) to automatically perform an operation when content arrives at a DIVA Core site. One of these operations is to restore the new content to a drop folder, which can then archive that content into another DIVA system. This functionality is similar to what you can achieve using AutoCopy. *See the DIVA Core SPM (Storage Plan Manager) User's Guide in the DIVA Core Documentation library for more information on SPM*.

DIVA Core itself provides the ability to transfer content from Servers to DIVA systems and back again, and copy and delete archived content.

DIVA Connect, DIVA Core, SPM, and DFM can work in unison to satisfy distributed media workflows.

2

Installation Planning

DIVA Connect is commonly configured at multiple DIVA sites. This chapter describes how to determine which DIVA Connect services to install and where. The major steps are:

1. Understand which sites to connect to achieve the desired workflows. See Understanding Site Connectivity.
2. Enable remote access for each site in the system (*or local access*). See Enabling Sites for Remote Access.
3. Configure client access for sites that have applications connected them. See Configuring Client Access.
4. Configure the access control and asset replication policies for the DIVA Connect network. See Configuring DIVA Connect Policies.

Understanding Site Connectivity

A DIVA Connect site is defined as exactly one DIVA Core installation (*which could exist in the cloud*) and one or more DIVA Connect services. Each site has a unique *sitename*. Each DIVA Connect service belongs to a particular site, indicated by the *LocalSitename* parameter in the DIVA Connect configuration files. A DIVA Connect site uses the *sitename* to refer to other sites in the network.

The most basic DIVA Connect connectivity uses DIVA Connect as a DIVA API proxy for a single DIVA Core system. In this configuration, you would create a DIVA Connect Direct Mode port in the ClientAdapter. This mode does not provide a view of multiple sites, and cannot be used (*for example*) to copy between sites. For more information on setting up DIVA Connect Direct Mode, see "Configuring Client API Ports" on page 4-1.

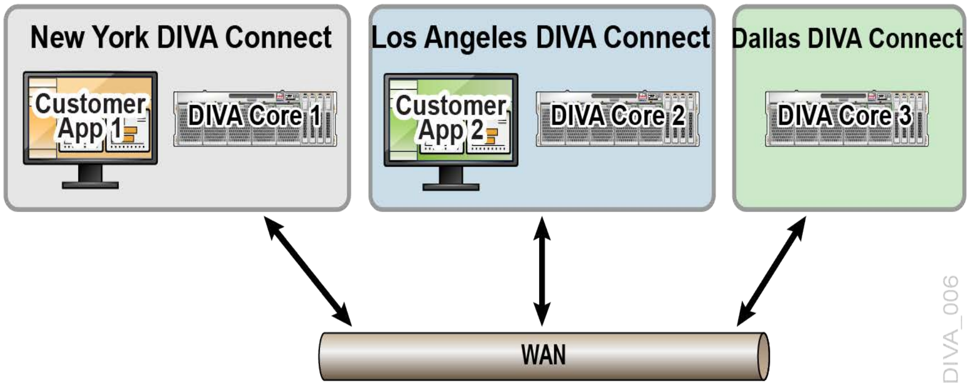
To have multiple DIVA sites act as one large archive system, DIVA Connect sites must be connected together using MultiDiva mode.

DIVA Connect can connect to remote sites to retrieve asset information, monitor the remote site's status, and send requests to the site (*for example, a restore request*), to satisfy DIVA Connect level requests. You can use DIVAConnectUI to view assets that are stored on multiple sites, copy assets from one site to another, and view DIVA Connect requests. You can use the DIVA Core API to connect to DIVA Connect and perform many of the same operations, and more.

> **Note:** Some DIVA Connect deployments will not require each site to be connected to all other sites in the network.

Sample DIVA Connect Deployment

The following figure shows an example of a typical DIVA Connect deployment with three sites: New York, Los Angeles, and Dallas. In this example, applications in New York can see and copy assets from Los Angeles and Dallas, as well as locally in New York. In addition, applications in Los Angeles can see and copy assets from New York and Dallas. No customer applications are running at the Dallas site.



DIVA Connect Services

A DIVA Connect service is either a Windows or Linux Service installed on a server responsible for carrying out computing tasks in a DIVA Connect deployment. Table 2–1 shows a summary of the available DIVA Connect services.

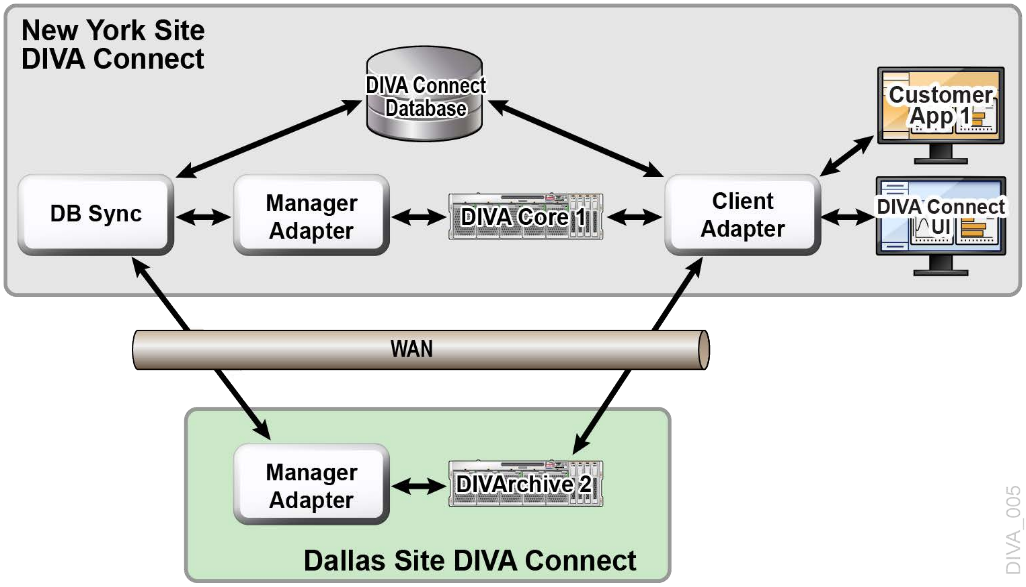
Table 2–1 DIVA Connect Services

Service	Description
ClientAdapter	Accepts requests from the DIVA API and web clients, and interacts with DIVA Core sites and the DIVA Connect database to satisfy those requests. You configure this service when implementing client application access. You can also use this in a minimal proxy-only DIVA Connect deployment (*DIVA Connect Direct Mode, described in* Configuring Client API Ports). For more information, see DIVA Connect ClientAdapter Service.
ManagerAdapter	Serves as a bridge between DIVA Connect and the DIVA Core Manager. This service provides remote access for a DIVA site. You must configure this service for all DIVA Connect sites that will be accessed by other sites remotely, especially those that will have asset information synchronized. For more information, see DIVA Connect ManagerAdapter Service.
DbSync	Responsible for synchronizing asset information from multiple DIVA Core sites, and storing the information in the DIVA Connect database. You configure this service when implementing client (*application*) access. For more information, see DIVA Connect DbSync Service.

Enabling Sites for Remote Access

To enable a DIVA Core site for remote access by other DIVA Connect systems, you must set up a ManagerAdapter service on the site and configure DIVA Core for remote access.

The following figure shows an example of two sites: a New York site with a full DIVA Connect configuration (*remote access and client access*) and a Dallas site that is configured just for remote access. The Dallas site only has one DIVA Connect service running: the ManagerAdapter service. DIVA Core has been configured so it can interface with other sites.



DIVA Connect ManagerAdapter Service

The ManagerAdapter service serves as a bridge between DIVA Connect and the DIVA Core Manager. It must be configured to provide remote access by other DIVA Connect systems.

For security and performance reasons, Telestream recommends that the ManagerAdapter be installed on the same system as the DIVA Core Manager. Likewise, it is often the case that the ClientAdapter and DIVA Connect database run together on a different server entirely.

The ManagerAdapter is configured using a simple configuration file. For more information, see Chapter 4.

DIVA Core

To achieve DIVA Connect workflows, you perform most of the configuration at each DIVA Core site. This section details some concepts needed to understand how DIVA Connect interacts with DIVA Core, and the importance of the DIVA configuration. *For the details on how to configure DIVA Core, refer to the DIVA Core Installation and Configuration Guide*.

Objects and Instances

In a DIVA Core system, archived objects are uniquely identified by two parameters: an Object Name and an Object Category. The Category is part of the formal name of the object, a kind of namespace. For example, an object with a name of *CLIP01* and category of *MOVIES* is a different object than one with a name of *CLIP01* and a category of *COMMERCIALS*.

DIVA Connect uses the object name and category to associate objects on various sites.

Note: If an object on one site has the same object name and category as on another site, DIVA Connect will consider both to be the same object.

When you archive assets using DIVA Connect, it will (*by default*) reject new assets that have the same name (*and category*) as assets already archived on other sites. However, archives issued directly to a DIVA Core system are not checked this way. Archiving without using DIVA Connect may result in an object on site B that has different content than the corresponding object on site A, event though the two objects have the same name and category. This can lead to DIVA Connect restoring the wrong content.

In DIVA Core, each archived object can contain many instances: one instance for each physical copy of the object on tape or on disk. There is an order number for each instance. The numbering starts with zero and increments by one for each instance in the object. So, you can uniquely refer to an instance on a DIVA system by providing the object name, category, and instance order number.

DIVA Connect assigns its own set of instance order numbers that are derived from the DIVA Core instance order number. It does this so that for each object, the DIVA Connect instance order numbers are unique across all DIVA Connect sites.

Source and Destination Servers

A DIVA Core Server contains information necessary to communicate with a customer server or disk external to DIVA Core. Customers transfer content to and from DIVA Core through these servers and disks.

If a Server on one site has the same name as one on another site, DIVA Connect concludes that they refer to the same physical server and (*or*) disk. This convention is important in setting up a DIVA Connect system (*see* Restore Workflow *for more information*). If the Servers are addressable through the API, and they point to the same physical server, disk, and path, you should give them the same name.

Setting Up Transfer Servers

To use DIVA Connect to transfer content from one site to another, configure at least one Server to be accessible from both sites. DIVA Connect will use this common Server to copy objects from one site to the other. The Server configurations on both sites should have the following characteristics:

- **Same Name** — On all sites, you should configure the same name for Servers that refer to the same physical server, disk, and directory.

  DIVA Connect's Site-To-Site Mappings can handle Servers that point to the same place but are not necessarily named the same. See Site-to-Site Mappings for more information.

- **Same Place** — The two Server entries must point to the exact same location (*path*) on the disk on a server. The types of transfer (*for example, FTP_STANDARD, DISK*) may differ on each site, and may even have different root paths in the configuration. For example, a Server named NY_SHOWS might be of type DISK on the New York site, but of type FTP on the Los Angeles site.

- **No Transcode or Renaming** — For Servers used in inter-site copies, do not configure the Server for transcoding upon restore. This will lead to incorrect content being archived to DIVA sites.

- **Delete on Source** — For each Server that will be used in copy commands, set the -allow_delete_on_source option in the DIVA Core Server configuration. This allows you to delete content from the site after it has been transferred to DIVA. You provide this option in the options field of DIVA's Server configuration panel.

- **AXF and Checksums** — You can enable end-to-end checksum comparisons on inter-site copies (*copy operations from one site to another*) by enabling ***AXF Genuine Checksums*** in DIVA Core. In the DIVA Core Configuration Utility, select the Server that you use for copies, and then select the ***AXF Genuine Checksum*** option. After this is performed, you can set the -axf option in the DIVA Connect Site-To-Site mapping *AdditionalOptions* parameter.

This allows checksum information to be embedded in the AXF wrapper on the Source Server site, and checked again at the target site.

- Punch Hole Server

  To use the Datahub to Datahub feature introduced, a Punch Hole Server must be installed and running on a server that is accessible to both the Source Server site and Destination Server site of intersite transfers. This server is used only by DIVA Connect for Datahub to Datahub transfers.

Do not be confused by the ***Site*** parameter in the DIVA Core Configuration Utility's Server panel. The sitename here is used only by DIVA, and does not correspond to a DIVA Connect site (*see the DIVA Core Installation and Configuration Guide for more information*).

> **Caution:** Modifying the names of DIVA Core configuration parameters (*such as **Source/Destinations**, **Media Names**, and **Storage Plans***) when connected to DIVA Connect can cause errors.

Media (*Storage Mediums*) and Storage Plans

When DIVA Connect copies objects from one DIVA system to another, take care when assigning the archive *Media Name* and *Storage Plan Name* of the copy on the target site. Use a good naming policy regarding media values on each DIVA system.

DIVA Connect records the DIVA Media Names when it synchronizes each object instance. You can configure DIVA Connect to automatically assign the Media/Storage Plan on a Copy operation. See Selected By DIVA Connect (media of **any)** for more information. One of the ways to configure this feature is to archive to the target site with the same Storage Plan Name as the source object. For this to work, the proper storage plans need to be configured in the target DIVA. Alternatively, you can use DIVA Media Mappings to transform the Storage Plan Name into a media or another storage plan, all at the target DIVA site.

DFM (*Drop Folder Monitor*)

DFM monitors folders for new content, and then archive the new content into DIVA Core. By restoring to a particular drop folder, DFM can pick up the content and archive it to a different DIVA system.

DIVA Connect can implement copy workflows without DFM, but in some cases, it is needed or desirable. To copy without DFM in the mix, the DIVA Connect RestoreAndArchive Transfer Method can be used. However, there are some situations where it is appropriate to use DFM. Good candidates for using DFM might include autonomous sites that want to perform their own cleanup of unsuccessfully transferred content, or systems where third party WAN accelerators are used. To use DFM for transfers, use the DIVA Connect RestoreAndMonitor Site-To-Site Transfer Method. See Site-to-Site Transfer Mappings (Workflow Profile) for more information.

Datahub to Datahub Transfer (*D2D*)

DIVA Connect 3.1.0 supports the Datahub to Datahub intersite transfer feature. You can configure two Datahubs on different sites to transfer content directly. This feature requires that the DIVA Bridge component of DIVA Core be setup and running, and must be configured in both the Source Server and Destination Server Core configurations. The name of the Bridge Server must be specified as the FromSrcDest and ToSrcDest in a SiteToSiteTransfer section of the *Site-to-Site Transfer Mappings*.

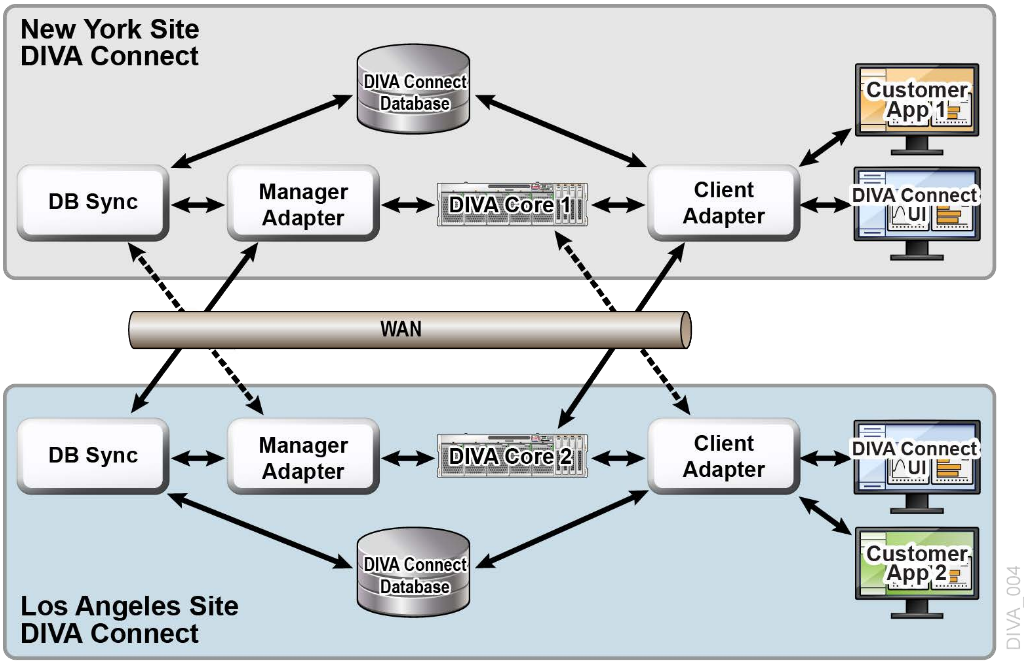
Configuring Client Access

Configuring for Client Access involves configuring:

- The local DIVA Core for remote access (*see* Enabling Sites for Remote Access)
- The ClientAdapter service
- A DbSync service
- A DIVA Connect database

Configuring all DIVA Connect services enables the site for full DIVA Connect workflow processing.

In the following figure, both the New York and Los Angeles sites are configured for full DIVA Connect workflow processing. Applications in Los Angeles connect directly to the ClientAdapter in LA. By doing so, they can retrieve content from New York if needed. The local DIVA Connect database provides a global view of assets across sites, even if connectivity is lost from one site to another. If sufficient permissions are granted, DIVAConnectUI users in LA can copy content from New York to LA, and even delete content in New York.



Lorem ipsum

Though it is technically possible to configure Customer App 2 to remotely connect to New York's ClientAdapter, this configuration often provides better availability, security, and auditing. Performance and scalability is often increased as well, especially with unreliable or slow WAN links.

DIVA Connect ClientAdapter Service

Application clients that want to use the DIVA API, or want to use the DIVA Connect GUI, connect to the DIVA Connect ClientAdapter service. This DIVA Connect service accepts web and socket connections from applications and processes the requests. A ClientAdapter is configured on each site that has applications that are *local* to the site where DIVA Core and DIVA Connect are installed. The ClientAdapter communicates with local and remote sites through the ManagerAdapter service. ClientAdapters can also connect directly to DIVA Core Managers using Socket Mode.

You configure the ClientAdapter service using one (*or two*) configuration files (*see* Chapter 4 *for more information*).

DIVA Connect DbSync Service

The DbSync service is responsible for synchronizing asset information from multiple DIVA Core sites, and storing the information in the DIVA Connect database. DbSync communicates remotely with ManagerAdapter services on multiple sites to synchronize archived object information. DbSync is typically deployed along with the ClientAdapter. Both the DbSync service and ClientAdapter require direct access to the DIVA Connect database.

You configure the DbSync service using a simple configuration file (*see* Chapter 4 *for more information*).

Display-Only Sites

You can configure a site as display-only, meaning that asset information from that site will be synchronized, but no requests (*or any other messaging*) will be sent to the site. Configure the site (*for example, site diva4*) in the DbSync configuration file, but not in the ClientAdapter configuration. Site *diva4* will be effectively display-only. Asset information from this site will be queryable in the UI and in informational API calls, but requests sent to the site (*using DIVA Connect*) will be rejected.

DIVA Connect Database

Configuring DIVA Connect local client access also involves setting up a DIVA Connect database.

Object Cleanup

DIVA Connect will sometimes satisfy a restore operation by temporarily copying an object from a remote site to the local site before restoring. This way, future restores of the content will be much faster. DIVA Connect does not automatically delete the disk instance after the restore. Instead, it leaves the content in case others will want to restore it.

DIVA Core contains two tools that can automatically clean up content when a given disk/array becomes full:

- DIVA Core SPM (*Storage Plan Manager*) has a feature that can automatically clean up disk instances for an individual DIVA site.
- DIVA Core Local Delete can perform a similar function, but can optionally verify that an object also exists on other DIVA sites.

Because DIVA Core is configured to create a nearline disk instance by default, object cleanup also may need to occur on a DIVA site configured for DIVA Connect remote access only.

DIVA Connect Version Compatibility

DIVA Connect 3.1.0 will interoperate with DIVA Core 7.3.1 or later. Some features incorporated into future releases of DIVA Core may not be accessible to DIVA Connect without upgrading DIVA Connect to a later release.

The DIVA Connect 3.1.0 ClientAdapter and DbSync services will interoperate with a DIVA Connect 3.1.0 ManagerAdapter only.

Configuring DIVA Connect Policies

DIVA Connect policies can be configured to enforce access control, automatically replicate assets, and delete assets over time. These policies stretch across sites, effecting how one site will access services or process information from other sites.

DIVA Connect allows rules to be defined that restrict certain operations in the API as well as in DIVA Connect UI. To configure access rules, see "Access Rules" on page 4-19. To set up users for authentication in DIVA Connect UI, see Adding Users in " DIVAConnectAdmin Options" on page 6-1.

DIVA Connect supports the ability to automatically replicate new assets to other sites. To define rules to automatically replicate assets to multiple sites, see "AutoCopy" on page 4-26. To defines policies governing how queued copy tasks will be scheduled and processed, see "Scheduling AutoCopy Tasks" on page 4-30.

The DIVA Core Local Delete application can remove assets from a site that are no longer needed. This service is installed separately from the DIVA Connect application, and has it's own configuration file. For more information, review the documentation for the DIVA Core Local Delete application.

3

Installing DIVA Connect

Before Installing

- DIVA Connect is a distributed application, meaning that it is often installed on multiple systems. Refer to Chapter 2 to understand what systems and servers DIVA Connect should be installed on, and which particular DIVA Connect services should be installed.
- Each DIVA Core site connected to DIVA Connect must be running DIVA Core 7.3.1 or later.
- DIVA Connect can be installed on 64-bit Windows platforms, or 64-bit Oracle Linux platforms (*release 7 or later*). Although you can install DIVA Connect in the same directory as DIVA Core, Telestream recommends that you install DIVA Connect in its own directory.
- On Windows computers the DIVA Connect installer requires administrator privileges. On Linux computers the DIVA Connect installer requires the user named diva.
- By default, the DIVA Connect installer handles the installation of the DIVA Connect services files, the DIVAConnectUI, and a compatible 64-bit Java 8 JRE (*Java Runtime Environment*). DIVA Connect uses the installed JRE for execution. Installing the DIVA Connect database requires an Oracle Database 11*g* release 2 or later (*Oracle 12c recommended*). Oracle SQLPlus (*Oracle 11 or later client software*) must be installed directly on the DIVA Connect system.
- If you are reinstalling DIVA Connect, make sure the DIVA Connect components are not running. The DIVA Connect 3.1.0 Windows installer has an option to automatically stop and start DIVA Connect services during installation.
- If a DIVA Connect database is installed, you will need to run a utility (see *Upgrading DIVA Connect*) to upgrade the database to the latest release. You can execute the *upgradeDb* functionality in DIVAConnectAdmin, on the command line, and the DIVA Connect Windows installer.

Installing DIVA Connect Software

DIVA Connect can be installed on either Windows or Oracle Linux 7 operating systems. The following subsections describe the installation on each platform.

DIVA Connect Windows Installation

Use the following procedure to install DIVA Connect in a Windows environment:

1. Launch the DIVA Connect installer executable, DIVAConnect_V3.1.0.version.exe using an account with administrator privileges.
2. Select the desired components, and then click **Next**.
   - Select ***Java*** if you have not previously installed in the target directory.
   - If you only want to install the DIVA Connect user interface, deselect ***DIVA Connect Services***.
3. Specify a target installation directory. This will be the DIVA Connect home directory.

> **Note:** You can use the DIVA Core installation directory as the DIVA Connect home directory. However, Telestream recommends installing DIVA Connect in its own directory.

4. If this is an upgrade of DIVA Connect, the installer will prompt if you would like to have DIVA Connect automatically backup the software, stop/start services, and perform database upgrades. If you select "y", and a database configuration exists, the installer will prompt for database connection information.
5. Click **Install**.

Oracle Linux DIVA Connect Installation

If you run on Linux, you must create a user named diva and a user named DIVA Connectsvc. The divaconnectsvc user must have permissions to write, read, and execute files owned by diva. The diva user must have sudo privileges for the Linux systemctl command. You must run the installation as the diva user, an admin user, or root. You must create the diva user before installing the software. An option is provided in the Linux installer package to create the diva and divaconnectsvc users before installation.

Use the following procedure to install DIVA Connect in a Linux environment:

1. Open a terminal session and navigate to the parent directory for the DIVA Connect release.
2. Place the DIVA Connect installation package (*for example, DIVAConnect-3.1.0-RELEASE.sh*) in the directory. The installer enables DIVA Connect to be installed in a child subdirectory of this parent directory.
3. Start the DIVA Connect installation using the following command:

   ```
   sh ./DIVAConnect-3.1.0-RELEASE.sh
   ```

   The install wizard provides an option to create the prerequisite diva and divaconnectsvc Linux users before installing the software. These users do not need to be recreated if they already exist and are assigned the correct permissions.

   The installation will not continue if the users do not exist. If the users already exist, or after you create them, select the option to install the DIVA Connect software.

4. When prompted for the installation directory, press **Enter** to use the default subdirectory named DIVA Connect, or specify a custom subdirectory name and then press **Enter** to create the installation directory.

   The DIVA Connect install process begins, and all files and folders are extracted into the specified target directory.

The following sections describe additional steps you may need to take to complete the installation. See Chapter 2 for more information.

Creating the DIVA Connect Database

To create the DIVA Connect database, you first create the DIVA Connect Oracle database user and tables, and then configure the DIVA Core sites. You can use the DIVAConnectAdmin utility to perform both of these activities. See Chapter 6, "DIVA Connect Administration" for more information. These tasks can also be executed via scripts, which are documented below.

Creating the Tables

This step involves creating the DIVA Connect database tables that will exist on a running Oracle server. This step is required only for a full DIVA Connect install (*see* Chapter 2).

Create the DIVA Connect Database Schema

To run the SQL creation scripts, you must have SQLPlus (*Oracle 11g or later client software*) installed on the system you are installing DIVA Connect on. Insure that the path to the sqlplus utility is in the PATH environment variable on your system.

1. Open a command line window or terminal shell. Change directory to the DIVA Connect home directory (*the directory where DIVA Connect is installed*).
2. Change directory to Program/Database/divaconnect/Install.
3. Run the following command:

   Windows:

   ```
   create_divaconnect_user.bat {sys_password} {user_name} {user_password} {oracle_tns_name}
   ```

   Linux:

   ```
   create_divaconnect_user.sh {sys_password} {user_name} {user_password}  {oracle_tns_name}
   ```

   In the previous command, *syspassword* is the Oracle system password, *user_name* is the Oracle user name/schema name that will contain the DIVA Connect tables, and *oracle_tns_name* is the Oracle TNS Name for the DIVA Connect database you want to use for installation. If there is no TNS Name defined for the database, you can refer to the following section to create a TNS Name real-time.

Deleting the DIVA Connect Database Schema

To delete the DIVA Connect database schema, execute the following steps.

Caution: This will delete **all** data in the DIVA Connect database.

1. Change directory to Program/Database/divaconnect/Install.
2. Run the following command:

   Windows:

```
delete_divaconnect_user.bat {sys_password} {user_name} {oracle_conn_string}
```

Linux:

```
delete_divaconnect_user.sh {sys_password} {user_name} {oracle_conn_string}
```

In the previous command, *syspassword* is the Oracle system password, *user_name* is the Oracle user name or schema name that will contain the DIVA Connect tables, and *oracle_conn_string* is either the Oracle TNS Name for the DIVA Connect database you want to use, or the Oracle Connection String (*see the following section*) for installation. If there is no TNS Name defined for the database, refer to the following section to create an Oracle connection string for use in the add and delete scripts.

Oracle Connection String

If you do not have a TNS Name defined for your database, you can supply an Oracle connection string. The format for this string is the following:

netaddress:port/remote_servicename

Where:

- *netaddress*: Network Address where the database runs
- *port*: Port where clients can connect
- *remote_servicename*: The Oracle remote service name. This is different from the service name used locally. The default for the DIVA Oracle database installation package is lib5.world. An example full connection string is 183.34.34.67:1521/lib5.world.

Adding Sites

After the DIVA Connect tables are created, you must configure the DIVA sites in the DIVA Connect database. You can execute this in DIVAConnectAdmin, or run the addSites script to configure the sites in the database as follows:

1. Change to the DIVA Connect home directory, and then navigate to the Program/divaconnect/bin subfolder.
2. Run the addSites script:

   Windows: addSites.bat

   Linux: addSites.sh

   This script will walk you through adding sitename records and configuring sitenames, locations, and comments. The network connections to the sites are configured in the ClientAdapter configuration file (*see* Chapter 4). You should create sitenames that are easy to read and reflect either the physical location, or the function of the site.

Note: The sitenames you provide in this step will be used when configuring both local and remote DIVA Connect services. Ensure the desired sitenames are consistent and agreed upon before running the addSites script. After the sitenames are in the database, DIVA Connect installation can proceed.

Configuring the DIVA Connect Configuration Files

Refer to Chapter 2 to determine which DIVA Connect services need to be configured, and then refer to Chapter 4 for details on how to configure each service.

Underneath the DIVA Connect home directory, in the Program/conf/divaconnect/templates subdirectory, there are template configuration files for each of the DIVA Connect services that need to be configured. Each template file has a .ini extension. You should take each file and copy it to the parent directory (*in this case, Program/conf/divaconnect*). When you copy each file, make sure to remove the .ini extension.

At the top of each .ini file, there is documentation of the parameters available. For each DIVA Connect site, configure a ManagerAdapterConfig.xml. For a full DIVA Connect installation, you also configure the ClientAdapterConfig.xml and DBSyncConfig.xml files.

Many of the parameters can be left at their default value. You should not have to change any of the configuration files in the Program/conf/divaconnect/wrapper folder. You may optionally add information to the divaEnv.conf file (*see* Adding Script Variables to a Configuration File *for more information*).

Note: If after reading this documentation you are unclear on what parameters should be modified from their defaults, contact Telestream Support.

Installing and Starting DIVA Connect Services

See Chapter 2 to determine what services need to be installed and configured.

After the configuration files have been set up properly, install and start the DIVA Connect services. Perform each step for every service to be installed at each DIVA Connect site.

The DIVAConnectAdmin utility can install DIVA Connect services, as well as start/stop them -- see Chapter 6, "DIVA Connect Administration" for more information. Scripts to perform these operations are also included, and are described below.

Installing DIVA Connect Windows Services

You can use DIVAConnectAdmin to install Windows services, or use the following procedure to install DIVA Connect services in the windows environment:

1. Change to the DIVA Connect home directory (*where DIVA Connect is installed*).
2. Navigate to the Program/divaconnect/bin folder.
3. Run the InstallDIVAConnectService.bat script, providing the wrapper configuration file for the service being installed. For example, to install the ManagerAdapter Service (*required at all sites where a Manager to be connected to DIVA Connect is running*), run the following command:

   ```
   InstallDIVAConnectService.bat ../../conf/divaconnect/wrapper/ManagerAdapterWrapper.conf
   ```

   - For the ClientAdapter Service, use the same command and same path, but substitute ClientAdapterWrapper.conf as the file name.
   - For the DbSync Service, use the file name DBSyncWrapper.conf.
4. In the Windows Services window, you should see one or more of the following DIVA Connect services:
   - DIVA Connect ManagerAdapter
   - DIVA Connect ClientAdapter
   - DIVA Connect DbSync

The services can be started in any order, but keep in mind that after the DbSync Service and the ManagerAdapter Services are both started, the database syncing process will start automatically. Start the ClientAdapter Service to allow new DIVA Connect connections and API processing.

Installing DIVA Connect Linux Services

You can install a Linux service using DIVAConnectAdmin, or use the following procedure to install DIVA Connect services in the Linux environment:

1. Open a terminal session and navigate to the Program/divaconnect/bin directory.
2. Run the divaconnectservice script to install the desired service (*DbSync, ClientAdapter, or ManagerAdapter*). For example, to install the ClientAdapter, enter the following command:

   ```
   ./divaconnectservice install ClientAdapter
   ```

Managing DIVA Connect Linux Services

After the Linux DIVA Connect services are created you use the following commands to manage them:

Table 3–1 Linux DIVA Connect Services

Operation	Command
List All Services	./divaconnectservice **list**
Start a Service	./divaconnectservice **start** {service_name}
Stop a Service	./divaconnectservice **stop** {service_name}
Restart a Service	./divaconnectservice **restart** {service_name}
Install a Service	./divaconnectservice **install** {service_name}
Uninstall a Service	./divaconnectservice **uninstall** {service_name}

The service_name parameter can be DbSync, ClientAdapter, or ManagerAdapter. The DIVA Connect services can also be managed using the standard Linux system V systemctl command syntax. You can enter man systemctl at the Linux command prompt for more information on the systemctl command.

Viewing Logs

The DIVA Connect applications produce two main types of logs:

- **Trace Logs** — Located within the DIVA Connect home directory, in the Program/log/divaconnect subfolder. Each of the DIVA Connect services (*ManagerAdapter, DbSync, and ClientAdapter*) create their own subfolder underneath the Program/log/divaconnect folder and generate their log files in that subdirectory.
- **Wrapper Service Logs** — Examine these first to make sure the applications are starting correctly. These logs are within the DIVA Connect home directory in the Program/log/divaconnect subfolder. They will be named similar to the Wrapper.conf file, but with a .log file name extension (for example: *ManagerAdapterWrapper.log*).

Checking the Configuration

The most common errors in configuring DIVA Connect are errors in specifying the sitenames. All DIVA sites in the DIVA Connect network must use the same set of sitenames. The sitenames must be consistent in all of the configuration files, and in the DIVA Connect database. The *LocalSitename* parameter must be correctly configured in all the service configuration files.

You should also check the DIVA Core Server. For example, by default, if site *diva1* has a Server that is named the same as one on site *diva2*, DIVA Connect considers them as pointing to the same logical Server. If the same Server name is configured on two different DIVA Core sites, DIVA Connect will assume that these Servers refer to the same place.

Syncing the DIVA Connect Database

When the DbSync service is started, it will attempt to synchronize archived object information by consulting the ManagerAdapter service. The ManagerAdapter service in turn pulls asset information from the DIVA Core site to which it is connected. Therefore, you should start the DbSync service and ManagerAdapter services only when you are ready to begin synchronization.

Note: If you stop the DbSync service, synchronization will start again where it left off when the service is restarted.

The DIVAConnectAdmin command-line tool monitors the status of the database synchronization on a site-by-site basis. If DIVA Connect is performing an initial synchronization of a particular site, it will show the percentage complete of that initial sync operation. If the site is up-to-date, DIVAConnectAdmin will indicate that the site is *In Sync*.

DIVAConnectAdmin can also be used to resynchronize a site. This involves deleting all archived object records previously synchronized from the site and resynchronizing them from the Source Server. This should only be used in rare circumstances, such as when the Source Server DIVA database has been unexpectedly restored from an out-of-date database backup.

You can run the gather_schema_stats script after an initial synchronization, or resynchronization, to immediately improve performance. *See the following section, Upgrading DIVA Connect, for more information*.

Upgrading DIVA Connect

It is recommended that you back up the existing DIVA Connect home directory before attempting to upgrade to a new version of DIVA Connect. You must also stop any running DIVA Connect services or programs on the server before attempting an upgrade. The new DIVA Connect 3.1.0 Windows Installer has an option to automatically backup the existing DIVA Connect directory, stop running services, and run the database upgrade utility. On Linux, use the following procedure to upgrade the DIVA Connect software:

1. Follow the instructions in the Installing DIVA Connect Software section. This will install the latest DIVA Connect software into the DIVA Connect home directory.
2. You must run the DIVA Connect database upgrade utility if you are upgrading a system that contains a DIVA Connect database. To run the utility, change directory to Program/divaconnect/bin and run the following command:

   Windows: ./upgradeDb.bat

   Linux: ./upgradeDb.sh

The script will prompt for input about the DIVA Connect database (*including database user name and the network address of where the database resides*). The utility will add or modify the required database objects. The script will simply return success if the database has already been upgraded to the latest release.

3. If you are upgrading a DIVA Connect 2.1 system that has a DIVA Connect database, you must run the gather_schema_stats script. This script installs a weekly job to gather statistics for the database (*if needed*), and then immediately gathers current statistics. The script requires the Oracle sys password. To run the script, change directory to Program/Database/divaconnect/Install, and execute the following:

   In Windows:

   ```
   ./gather_schema_stats.bat {divaconnectDbUser} {sysDbPassword} {oracleConnectionStr}
   ```

   In Linux:

   ```
   ./gather_schema_stats.sh {divaconnectDbUser} {sysDbPassword} {oracleConnectionStr}
   ```

   For example, in Linux execute:

   ```
   ./gather_schema_stats.sh divaconnect sysDbPass3 127.0.0.1:1521/lib5.world
   ```

4

Configuring DIVA Connect Services

Before configuring DIVA Connect services, see Chapter 2 for a description of the DIVA Connect services and how they interact with DIVA Core and each other.

- Configuring the ClientAdapter Service
- Configuring the ManagerAdapter Service
- Configuring the DbSync Service
- Access Rules
- AutoCopy
- Adding Script Variables to a Configuration File

Configuring the ClientAdapter Service

Configuring the ClientAdapter service involves configuring how clients connect to DIVA Connect. It also involves configuring how DIVA Connect connects to DIVA Core.

Configuring the DIVA Core Sitename

A DIVA Connect site is defined to be one DIVA Core System, and one or more of the DIVA Connect services (*ManagerAdapter, ClientAdapter, DbSync*). Each site is assigned a unique sitename. The site names must be configured in the DIVA Connect Database (*either DIVAConnectAdmin or addSites can be used to accomplish this*) before configuring the ClientAdapter. You should create sitenames that are easy to read and reflect either the physical location, or the function of the site. These sitenames will be used on each site.

Note: Sitenames are case sensitive.

Configuring Client API Ports

DIVA Connect allows configuration of the socket ports that client API applications use to connect to DIVA Connect. The ClientAdapter allows multiple API ports to be configured. Each API port is configured in one of two modes: Direct mode or MultiDiva mode. Each port can be assigned a Workflow Profile.

- **Direct Mode** — In Direct mode, an incoming API connection is routed directly to an individual DIVA Core system with little added processing. The particular site that DIVA Connect routes to is configured in the ClientAdapter on a per-port basis. This functionality allows local clients to connect to a remote DIVA Core system as if it were local. The DIVA site to route messages to must be configured in the DivaManagers section of the

ClientAdapter configuration file. Note also that a site can be configured to operate in Direct Mode only (*see Configuring the DIVA Core Sitename for more information*).

Requests in Direct mode will not be displayed in DIVAConnectUI, and can only be monitored at the DIVA site where the requests are being routed.

Some DIVA Connect workflows do not require a DIVA Connect database, DIVA Connect UI, or MultiDiva mode processing. To configure a ClientAdapter service for Direct Mode only, insure that only Direct Mode ports are defined, and that a WebServicePort is not defined. This effectively disables use of the DIVA Connect UI. An example configuration can be found in the file found here:

Program/conf/divaconnect/templates/ClientAdapterConfig.xml.ProxyOnly.ini

- **MultiDiva Mode** — In MultiDiva mode, DIVA Connect makes all DIVA sites appear to be one big archive system. In this mode, an incoming API request is routed directly to DIVA Connect. DIVA Connect satisfies the high-level request by consulting the DIVA Connect database and other DIVA Core systems as needed. MultiDiva Mode allows (*for example*) copying of content from one site to another, restores without needing to know which particular site has the content, and retries on other sites when content on the first site is inaccessible. In addition, it provides a global view of content on all sites.

  The progress of a request executed in MultiDiva mode can be monitored in the DIVAConnectUI. DIVA Connect will create its own request-level events that inform the user of:

  - How DIVA Connect will process the request
  - Requests that are made of DIVA sites
  - Any errors or warnings encountered along the way

Configuring Client Web Connections

The DIVAConnectUI application and the DIVAConnectAdmin tool both make web connections to the ClientAdapter. You can configure the port that is made available for these connections in the ClientAdapter configuration file.

Configuring the REST API

The REST API can be used to send requests and commands to DIVA Connect beginning with release 3.1.0. The same port used for the Client Web Connections (*previously mentioned*) is used for this API. The REST API supports only the *MultiDIVA Mode* as previously defined.

Configuring Workflow Profiles

A Workflow Profile is a set of parameters that defines how incoming requests will be processed by DIVA Connect. Workflow Profiles provide a way to logically group users and systems that use DIVA Connect in a similar manner.

DIVA Connect allows one or more profiles to be created within the ClientAdapter. These profiles contain parameters that are needed by a particular set of users, or set of applications. Workflow Profiles are assigned based on where the request was received (*the port where it was received - see the following section*).

In the Workflow Profile you can customize information such as the list of accepted valid messages, retry parameters, site-to-site copy methods, and other attributes.

Profiles and API Ports

A Workflow Profile can be selected by configuring the port that the DIVA API is connected to, by using a profile configured for a user, or by using the default (use ***default** for API users and a configurable value for UI users*).

In the API Ports section, each defined port can be assigned a Workflow Profile Name. Requests received on the API port are processed using the assigned Workflow Profile. If no name is defined, the profile **default** is assumed. Ports in MultiDiva mode can have Workflow Profile Names that refer to a Workflow Profile body section further down in the ClientAdapter configuration.

If the port is in Direct mode, the Workflow Profile Name is a label only - there is no Workflow Profile body for Direct mode connections. This name, however, can be useful in Access Rules (*see the following section*).

In both MultiDiva mode and Direct mode, if you do not assign a name, the connection uses a Workflow Profile named **default**. Each workflow profile body section is configured in the ClientAdapterConfig.xml. The following sections list the types of information configurable in each Workflow Profile.

Retries and Timeouts

Some commands within DIVA Connect can be retried (*for example, Restore, Copy, and Delete requests*). The parameters for the retries, including how long to retry and the retry interval are configurable here. Many commands have their own retry settings (*for example, the Copy command retry parameters are completely distinct from the Delete command retry parameters*). Message timeouts and connection limits can also be configured in the Workflow Profile. The retry and timeout settings are specific to the Workflow Profile in which they are defined.

Valid Messages

In each Workflow Profile, a list of messages is configured. This represents the list of valid API messages that can be accepted by this Workflow Profile. For example, if the Archive message does not appear in the list, then Archive messages cannot be sent to DIVA Connect through the API (*at least for this Workflow Profile*).

Site-to-Site Mappings

DIVA Connect provides a flexible way to configure how Site-to-Site transfers are performed. In each Workflow Profile, you configure mapping parameters for each transfer *path* (*one path would be transfers from site A to site B, another would be from site A to site C, and so on*). This table is consulted when Copy, Restore, or Partial File Restore commands are received.

It is useful to define the Site-To-Site mappings in the **default** Workflow Profile, as other Workflow Profiles can load their mappings from the **default** Workflow Profile. This can reduce the number of mappings in the configuration.

Parameters — DIVA Connect supports several different transfer types. Site-to-Site Transfer Methods describes the different types. **RestoreAndArchive** is the default transfer type. **RestoreAndMonitor** requires DFM (*Drop Folder Monitor*), or another application that performs a similar function. ActorSiteTransfer uses the Datahub to Datahub feature and requires the Hole Punch Server. Other parameters include:

- *Server* — the common storage location for transfer (*accessible at both the source and target servers*).
- *Default Media* — the default archive media to use when archiving on the target site. Default Media can be used when either a temporary copy was initiated (*such as by a*

restore), or a user launched a copy and indicated that DIVA Connect should select the media.

- *Options* — the parameters to use in the restore, archive, and transfer requests.
- *FilePathRoot* — the parent directory to store the content

Directory Location — DIVA Connect constructs a directory path to store the files that is relative to the Server chosen. This relative path is constructed as follows:

{FilePathRoot} / {Media} / {UniqueDirName} /

The *FilePathRoot* is specified in the Site-To-Site mappings. Media will be embedded in the path only if the *AppendMediaToPath* option is set to true in the mappings (*the default is **false***). Finally, DIVA Connect generates a unique directory name that is included as part of the path. This unique name has the sitename that initiated the request prepended to it.

Reloading Workflow Profiles

Parameters specified in the Workflow Profile can be changed and reloaded without restarting the ClientAdapter. Restarting the ClientAdapter service should be avoided, because it will stop all executing requests and close all client connections. The reload can be performed within the DIVAConnectAdmin tool (Chapter 6, "DIVA Connect Administration" *for more information on DIVAConnectAdmin*).

Note: The reload updates not only all of the Workflow Profiles, but also all of the Access Rules and AutoCopy configuration.

Modifying the ClientAdapter Configuration File

The following tables describe the parameters that can appear in a ClientAdapter configuration file. The file is in XML format. The default value column indicates the value that the parameter will have if the parameter is not specified in the configuration file. The column also indicates whether the parameter is required or optional.

To create a new configuration file:

1. Navigate to the DIVA Connect home directory (*where DIVA Connect is installed*).
2. Navigate to the Program/conf/divaconnect/templates folder.
3. Copy the ClientAdapterConfig.xml.ini to the parent directory, but leave out the .ini extension (*../ClientAdapterConfig.xml*).
4. Using the tables below as a reference, edit the parameters in the ClientAdapterConfig.xml file to configure the ClientAdapter.

See Appendix A for a sample ClientAdapter configuration file.

Top Level Parameters

Table 4–1 ClientAdapter Top Level Parameters

Parameter	Description	Default Value
LocalSitename	The sitename of the Local DIVA Connect site.	None (*required*)
LogDirectory	The folder where the log files will be produced.	The log/divaconnect/ClientAdapter directory

Table 4–1 (Cont.) ClientAdapter Top Level Parameters

Parameter	Description	Default Value
LogLevel	The detail level at which trace file logging takes place (***ERROR, WARN, INFO, DEBUG, TRACE***).	INFO
SyncTimeoutSecs	The number of seconds to wait for an object to synchronize.	60
WorkerThreads	The number of worker threads in DIVA Connect thread pools. Used for tuning large configurations. If you are unsure, do not set this value.	25
AbortAllOnStartup	Stop all incomplete DIVA Connect requests when DIVA Connect starts - even if the request has completed at the DIVA Core level. New requests are not affected.	false
MaxClientConnections	The maximum total number of API connections allowed.	200
GlobalDivaconnectRequestLimit	The maximum number of pending or running DIVA Connect requests that can be accepted into the system. When this limit is reached, DIVA Connect will start rejecting new requests.	5000
InternalPollingRateMillis	The base rate at which sites are polled for information. Change this parameter only if slow networks and (*or*) systems are encountered.	4000
WebServicePort	The port used to send management messages to the ClientAdapter.	None (*optional*)
SSLWebServicePort	True if SSL should be enforced for web service requests.	true
WebDefaultWorkflowProfile	Workflow profile to be used for web requests (*including DIVAConnectUI*).	The **default** profile
LogFileLifetimeHours	Number of hours to keep log files. Files older than this will be removed at the top of the hour.	50
apiSessionTimeoutMins	Lifetime of the API session token. This should not be changed without consulting Telestream Support.	1400
MaxNetAddressesPerUser	The maximum number of network addresses per user	2
UserAuthEnabled	If true, all DIVA Connect UI users are authenticated against user info in the database	false
AutoCopyRulesFilename	The name of the AutoCopy rules configuration file	None (*AutoCopy disabled*)

Table 4–1 (Cont.) ClientAdapter Top Level Parameters

Parameter	Description	Default Value
AccessRulesFilename	The access rules file name. This file name is relative to the directory where the client adapter configuration file exists.	None (*if not configured, no access rules will be applied*).

API Ports Section

Inside of the ApiPorts tag, multiple ApiPort definitions can appear. Table 4–2 shows the parameters that can appear as part of an ApiPort definition.

Table 4–2 API Port Parameters

Parameter	Description	Default Value
ListenPort	This is the port socket to listen on.	None (*required*)
RoutingMode	This identifies how to route requests (*Direct or MultiDiva*). • Direct — Route to one Manager only (*the Sitename parameter is required in this case*). • MultiDiva — Route using DIVA Connect Workflow Commands. Requests submitted will receive a unique Request ID by DIVA Connect.	MultiDiva
Sitename	The site to route to if using Direct Mode. Sites are defined in the DivaManagers section (*covered in the following table*). This parameter is only required in Direct Mode. If defined, it must match one of the sitenames defined in the DivaManagers section.	None (*required in Direct Mode*)
LocalAddress	The local address used to send to this Manager (*typically which network card to use*). Do not specify this if you're not sure.	None (*optional*)
SSLSocket	Enables SSL encryption on the API socket connection (*requires DIVA API 7.6 or greater to use*).	False
WorkflowProfile	The Workflow Profile Name to use for requests traveling through this port (*see the Workflow Profile section*). If it is not supplied, the **default** Workflow Profile will be used.	**default** (*the default profile in MultiDiva Mode*).

DIVA Core Managers Section

Inside of the DivaManagers tag, multiple DivaManager definitions can appear. Table 4–3 shows the parameters that can appear as part of an DivaManager definition.

Table 4–3 DivaManagers Configuration Parameters

Parameter	Description	Default Value
Sitename	The name of the site where the Manager is installed. The sitename listed must match what was configured in the DIVA Connect database and what is configured the ManagerAdapter.xml file.	None (*required*)

Table 4–3 (Cont.) DivaManagers Configuration Parameters

Parameter	Description	Default Value
ConnectionType	This identifies how to connect to the Manager (*valid values: **Socket, WebService***).	Socket
Address	The Network Address (*IP or internet name*) of the Manager	localhost
Port	The port where clients connect to the Manager.	None (*required*)
LocalAddress	The local address used to send to this Manager (*typically which network card to use*). Do not specify this if you're not sure.	None (*optional*)
LocalPort	The local port used. Do not specify this if you're not sure.	0
BaseURL	The url of the ManagerAdapter if the *ConnectionType* is set to **WebService; required**.	None (*optional*)
TotalThrottleThreshold	DIVA Connect will wait until the total number of Manager requests falls below this limit before sending more requests. This is used when the *ConnectionType* is set to **Socket**. When the Manager exceeds the number of running requests regardless of source (*for example, DIVA Connect, SPM, local API connection*) DIVA Connect will not send any more requests until the number of running requests on the Manager is below this threshold value.	400
SubmittedThrottleThreshold	DIVA Connect will wait until the number of requests DIVA Connect itself has running on a Manager falls below this limit before sending more requests. When the Manager exceeds the number of running requests sent solely from DIVA Connect itself, DIVA Connect will not send any more requests until the number of running requests on the Manager sent only from DIVA Connect is below this threshold value. Zero is a valid value, causing all messages to be queued internally within DIVA Connect. The value -1 indicates there is no limit.	100

DIVA Connect Database Section

Table 4–4 shows the parameters that can appear within the DivaconnectDatabase section.

Table 4–4 DIVAConnectDatabase Parameters

Parameter	Description	Default Value
Address	The IP Address of the database.	localhost
Port	The port used to access the database.	1521
User	The user name of the schema.	None (*required*)
Password	The password of the schema.	None

Table 4–4 (Cont.) DIVAConnectDatabase Parameters

Parameter	Description	Default Value
DbSiteId	The Oracle SID. This value is deprecated in favor of DbServiceName.	lib5
DbServiceName	The Oracle Service Name. Can be supplied instead of DbSiteId.	None (*optional*)
SecureMode	A value of "1" enables SSL encryption of the database connection.	None (*optional*)
DbCreds	Contains the encrypted database credentials generated by DIVAConnectAdmin. This parameter cannot appear when Password is used.	None (*optional*)

Workflow Profile Section

Table 4–5 shows the parameters that can appear within a WorkflowProfile section.

Table 4–5 WorkflowProfile Parameters

Parameter	Description	Default Value
Name	The name of the Workflow Profile	default
AllowDirectRemoteRestores	Allow direct transfers (*to Servers*) from remote DIVAs. Set this to **false** to always create a local copy of content before restoring.	true
MessageTimeoutMillis	The default timeout for messages sent to Managers.	15000 (*15 seconds*)
TotalRequestTimeoutHours	How long to keep requests before timing them out (*in hours*).	72
PreventArchiveIfInDirectory	Prevent new Archive requests if the object exists on any site. If set to true, and the object exists on any site, requests to archive that object will be rejected (*even if the object is not present on the site you are archiving to*).	true
DeleteRetryIntervalMins	The interval between retries for the Delete Workflow	5 minutes
DeleteRetryLimitMins	The total number of minutes for which Delete Workflow retries will continue. Delete retries will be attempted every DeleteRetryIntervalMins, for DeleteRetryLimitMins, or until the delete is successful.	0 minutes (*do not retry*)
DeleteWaitTimeoutMins	The total number of minutes to wait for a delete operation to complete. Set this value low for real-time deletes, and higher for batch operations.	2 minutes
IntersiteCopyRetryIntervalMins	How long to wait before retrying a copy request. This parameter applies only for Copy requests.	5 minutes
IntersiteCopyRetryLimitMins	Keep retrying copies until this total time limit is reached (*or they succeed*). This parameter applies only for Copy requests.	0 minutes (*do not retry*)

Table 4–5 (Cont.) WorkflowProfile Parameters

Parameter	Description	Default Value
RestoreRetryAttempts	In the case of failure, the maximum number of retries that should be attempted (*commonly, retrying with content that exists on another site*).	3 times
RestoreRetryIntervalMins	The interval between retries when DIVA Connect performs a retry with the same site.	5 minutes
SiteDownRequeueWaitMins	(*Advanced*) Maximum time a site is down before routing queued requests to another site. Deletes are also governed by the DeleteWaitTimeoutMins parameter.	20
BackupArchiveSite	If the local site is down for an extended period (*configured in SiteDownRequeueWaitMins*), the site to use for archives instead of the local site. If the archive is submitted and fails on the local site, the archive will not be retried on the backup site.	None (*no backup site*)
ForceGlobalDeleteToSite	Converts a global delete request to a site delete at the specified site.	None (*optional*)
ViewOwnedRequestsOnly	If true, users with this profile can only view requests issued by a user with this profile.	False
Messages	One or more Message Lists, each having a list of valid messages for the Workflow Profile.	NA

Table 4–5 (Cont.) WorkflowProfile Parameters

Parameter	Description	Default Value
Message	One or more valid Message Names: • AllInfo • Archive • Cancel • CloseObjectsList (*legacy*) • Copy • Delete • DeleteFile • GetArchiveSystemInfo • GetArrayList • GetFilesAndFolders • GetGroupsList • GetObjectsByFilename • GetObjectDetailsList • GetObjectInfo • GetObjectsList (*legacy*) • GetRequestInfo • GetSourceDestinationList • GetStoragePlanList • InitObjectsList (*legacy*) • PartialRestore • Restore • TranscodeArchived • CopyToNew At least one Message tag must be specified. AllInfo allows all informational request messages to be sent.	None (*one required*)
UseDefaultMappings	True if the Workflow Profile should include all of the mappings defined in the **default** Workflow Profile.	False

Site-to-Site Transfer Mappings (*Workflow Profile*)

The Mappings tag contains multiple Site-To-Site Transfer mappings. A Site-To-Site Mapping defines how an object is copied from one site to another. Each mapping contains FromSitename and ToSitename parameters. Each mapping defines how copies are performed from FromSitename to ToSitename.

Each mapping contains a Type parameter that indicates the method used to perform the transfers (*see* Restore Workflow *for more information*). The other parameters are defaults that are used in the process of executing the intersite copy.

DIVA Connect will use the FromSrcDest as a temporary storage area, and will optionally push to the target DIVA using the ToSrcDest. When storing the content DIVA Connect provides a unique folder name that is appended to the FilePathRoot. After storing the content in the FromSrcDest, DIVA Connect will (*based on the Type parameter*) either:

- Archive the content into the target site
- Wait until the content is archived successfully at the target site
- Complete with no further action

> **Note:** To avoid specifying the same mappings multiple times in the configuration, you can set the Workflow Profile parameter UseDefaultMappings. The Workflow Profile will use mappings from the **default** Workflow Profile.

Table 4–6 SitetoSiteTransfer Workflow Profile Parameters

Parameter	Description	Default Value
FromSitename	Origin sitename where objects are copied from. The value entered must match one of the sitenames defined in the DivaManagers section.	None (*required*)
ToSitename	Destination Server sitename where objects are copied to. The value entered must match one of the sitenames defined in the DivaManagers section.	None (*required*)
Type	The type of transfer: • **Restore** — Perform Restore and mark as transferred. • **RestoreAndArchive** — Restore then Archive to the Destination Server DIVA Core. • **RestoreAndMonitor** — Restore, then monitor Destination Server (*useful for DFM drop folders*). • **ActorSiteTransfer** — Transfer using the Datahub to Datahub feature.	
FromSrcDest	The Server to use in the Restore step of the Copy.	
ToSrcDest	The Server to use in the Archive step of the Copy.	
TempDefaultMedia	The target media to assign when making a temporary or transient copy of the object (*performed on some Restore operations*). The value is also used (*depending on configuration*) when API or UI users want DIVA Connect to decide which media to use (*the **any** keyword is used as the media*). When **RestoreAndMonitor** is used with this variable, decide if the AppendMediaToPath option is needed.	None (*required for **RestoreAndArchive***)
FilePathRoot	A path segment relative to the Server root. Prefixed to the unique folder name generated by DIVA Connect.	Remote
AdditionalOptions	DIVA Options to use in the Restore/Archive operations.	**-axf 1.0 -rm -delete_fpr** **-allow_delete_on_source**

Table 4–6 (Cont.) SitetoSiteTransfer Workflow Profile Parameters

Parameter	Description	Default Value
AssignDefaultMediaOption	The strategy to use when API or UI users decide to let DIVA Connect choose which media to use for copies. This is invoked when the **any** keyword is used as the media. **StoragePlan** — Use the Storage Plan name of the source object as the default media. **StoragePlanAndSitename** — Prepend Source Sitename to the Storage Plan (*separated by an underscore*). **TempMedia** — Use the value of TempDefaultMedia as the media.	**TempMedia**
AppendMediaToPath	True if the target media should be appended, as a subdirectory, after the FilePathRoot (*and before the unique folder name*). You would end up with the following: FilePathRoot / ToMedia / UniqueFolderName This option is useful when using the **RestoreAndMonitor** type with DFM, as DFM can parse the media name passed in this way.	false
Weighting	Rate this transfer path against other transfer paths based on performance, preference. Used in choosing sites for Copy and Restore. Valid range is 0-40. Be careful in using values above 20, as you will start to override other Datahubs such as disk versus tape, site status, and so on. **Misuse of this option can cause performance issues on Restore operations, and can contribute to WAN network congestion.**	Default is 10, local gets increased by 10.

Preferred Server Mappings

When DIVA Connect calculates which site to use for Restore operations, DIVA Connect typically prefers the local site, provided the Server is reachable using the local site. However, there are some cases where another site may be preferable.

The Mappings tag can contain a SrcDest tag. Within the SrcDest tag, a Name tag defines a Server name. The PreferredSitename tag indicates the preferred site to use when the Server is requested in a Restore operation. Multiple SrcDest sections can be present.

Using the Weighting Parameter

If the restore site has not been provided in the request, the preferred site is chosen by calculating a score for each site. Scores are calculated by adding points assigned as follows:

- Manager Status
    - Manager is running nominally: 60 points
    - Manager is running, but the request queue is full: 40 points
    - Manager is down: 0 points
- Is the local site: 20 points
- The site has a disk instance: 30 points
- Can the site perform a direct restore: 20 points
- Weighting as assigned by configuration: Default is 10 points

Example:

The local and remote sites are the same with the exception that the remote site has a disk instance and the local has only tape instances. The local site will get a score of 60+20+20+10=110, and the remote site will score 60+30+20+10=120. The remote site will be selected. This may be undesirable if the remote is a cloud site, and a cost is invoked. To force the selection of the local site under these conditions, set a Weighting value of 21 to the local site by including the following in the Mappings section:

```
<SiteToSiteTransfer>

<FromSitename>[local site name]</FromSitename>

<ToSitename>>[local site name]</ToSitename>

<Type>Restore</Type>

<Weighting>21</Weighting>

</SiteToSiteTransfer>
```

Configuring the ManagerAdapter Service

The ManagerAdapterConfig.xml configuration file contains the configuration for the ManagerAdapter Service. Use the following steps along with the description of each parameter in the ManagerAdapter configuration (*each of the following tables has this information*) to configure the ManagerAdapter Service.

Sync Filtering By Category

DIVA Connect can place a filter on object information that is pulled by the DIVA Connect DbSync service. This filter enables a site to select the subset of object records to be synchronized with downstream DIVA Connect systems. The filtering is configured in, and performed by, the ManagerAdapter service.

Note: Object Filters and Category Prefix Substitution are advanced features that require careful testing to implement in production workflows. Do not add or change object filters indiscriminately.

For example, DIVA Connect system in New York is configured to use and store assets from a site in Los Angeles. The Los Angeles site administrator wants to ensure that users in New York will only see objects that match one of three categories, AVID, POST1, and POST2. Implementing the following filter through the Los Angeles site's ManagerAdapter configuration file accomplishes this:

```
<LocalSitename>LosAngeles</LocalSitename>
<ObjFilter>
   <RequestingSitename>NewYork</RequestingSitename>
   <Category>AVID</Category>
   <Category>POST1</Category>
   <Category>POST2</Category>
</ObjFilter>
```

With this filter, objects matching the specified categories will be synchronized to the DIVA Connect database in New York. Objects with other categories will not be synchronized. From the perspective of a New York site user, the only object records that exist on Los Angeles are the objects matching the configured category filter. Multiple ObjFilter tags may appear in the ManagerAdapter, each with a set of categories specific to a requesting site.

> **Note:** Category Object Filters do not automatically prevent the ManagerAdapter from accepting requests for objects not matching the filter. ManagerAdapter Access Rules prevent operations on objects that do not have certain categories.

To prevent operations on objects not matching the filter, create the following access rule in the ManagerAdapter Access Rules file:

```
<Include>
   <SourceSitename>NewYork</SourceSitename>
   <Operation>*</Operation>
   <ReqObjectCategory>AVID</ReqObjectCategory>
   <ReqObjectCategory>POST1</ReqObjectCategory>
   <ReqObjectCategory>POST2</ReqObjectCategory>
</Include>
```

This rule will only permit requests from New York for objects that match one of three categories, AVID, POST1, and POST2. Other categories are denied if no other Include Rules are specified. If you define access rules in the ManagerAdapter, ensure that the ClientAdapter is configured to communicate with the site in *WebService* mode.

Configuring Category Prefix Substitution

There is a potential problem when using sync filtering by category. Continuing with the example, if New York creates an object with a category that is not present in the filter, and copies this object to Los Angeles, a naming conflict could occur. An object with that name could already exist on Los Angeles because the DIVA Connect system in New York does not know about those objects. One solution is providing Access Rules in New York that limit the potential categories that can be archived.

You can achieve a more flexible solution using category prefix substitution. This feature not only filters the entries that are synced, but adds a category prefix to each incoming request. This provides a namespace-like feature for archived objects at a site.

In some DIVA Connect workflows, a single site must accept objects copied from multiple sites. This can make it difficult to establish a set of unique categories for all objects in the system. Use category prefix substitution to address this situation. The following ManagerAdapter object filter will ensure that only objects in Los Angeles with categories beginning with NY001. are synchronized to New York's DIVA Connect database.

```
<LocalSitename>LosAngeles</LocalSitename>
<ObjFilter>
   <RequestingSitename>NewYork</RequestingSitename>
   <CategoryPrefix>NY001.</CategoryPrefix>
</ObjFilter>
```

After applying the filter, but before the object reaches the Destination Server (*New York*), the prefix is stripped – the remaining characters serve as the category in the New York DIVA Connect database. For example, if the category of an object in Los Angeles is NY001.POST1, the resulting category sent to New York will be POST1. Similarly, any time the New York DIVA Connect ClientAdapter sends commands to Los Angeles, the prefix is added back.

This allows Los Angeles to store copies of all of the objects from New York without naming conflicts. This technique allows Los Angeles to serve as a Disaster Recovery site for multiple sites. New York does not need to change its naming policy. New York references objects the same way that it always has, therefore no renaming on New York is necessary. Only one category prefix is allowed for each requesting site.

For this to work, the DIVA Connect ClientAdapter service must be configured to connect to the remote site in *WebService* mode. Because objects are essentially renamed when copied to the site with a prefix substitution enabled, objects that were copied to the site previously will not have the prefix, which can present an issue. One solution involves providing a list of additional categories that are not translated. Another solution involves having DIVA installation specialists rename a subset of objects in the filtered site (*that is, adding the category prefix to the category of each affected object in the DIVA database*). If you use Category Prefix Substitution, you (*will likely need to*) disable Category checking in the DIVA Core Datahub configuration (*contact Technical Support for instructions*).

Note: If you change an Object Filter, it will almost always be necessary for the downstream DIVA Connect system to perform a resync of the site. That system can accomplish this by using the DIVAConnectAdmin tool (*refer to* Chapter 6).

The two types of category filtering can be mixed together. The following second filter (*to Dallas*) performs category prefix substitution (*using DAL01*) on all categories except the categories POST2 or POST3. Only object records that have the category prefix or have a category of POST2 or POST3 will be synchronized to Dallas.

```
<LocalSitename>LosAngeles</LocalSitename>
<ObjFilter>
   <RequestingSitename>NewYork</RequestingSitename>
   <CategoryPrefix>NY001.</CategoryPrefix>
</ObjFilter>

<ObjFilter>
   <RequestingSitename>Dallas</RequestingSitename>
   <CategoryPrefix>DAL01.</CategoryPrefix>
   <Category>POST2</Category>
   <Category>POST3</Category>
<ObjFilter>
```

If you use this hybrid approach, ensure that object names with categories appearing on the list (*for example POST2*) are not added again with the prefix (*for example NY001.POST2*). This policy can be enforced through Access Rules.

Modifying the ManagerAdapter Configuration File

The following tables describe the parameters that can appear in a ManagerAdapter configuration file. The file is in XML format. The default value column indicates the value that the parameter will have if the parameter is not specified in the configuration file. The column also indicates whether the parameter is required or optional.

To create a new configuration file:

1. Navigate to the DIVA Connect home directory (*where DIVA Connect is installed*).
2. Navigate to the Program/conf/divaconnect/templates folder, copy the ManagerAdapterConfig.xml.ini to the parent directory, but leave out the .ini extension (*../ManagerAdapterConfig.xml*).
3. Using the tables below as a reference, edit the parameters in the ManagerAdapterConfig.xml file to configure the ManagerAdapter.

A sample ManagerAdapter configuration file appears in Appendix A.

Table 4–7 Top Level ManagerAdapter Parameters

Parameter	Description	Default Value
LocalSitename	The name of the local site. The sitename listed must match what has been configured in the DIVA Connect databases (*local and remote*), and must match what was configured in both the ClientAdapterConfig.xml and DBSyncConfig.xml files. This configuration enables the ClientAdapter and DbSync to communicate with the ManagerAdapter.	None (*required*)
ManagerAddress	The network address (*IP or internet name*) of the DIVA Core Manager.	localhost
ManagerPort	The port where clients connect to the DIVA Core Manager.	None (*required*)
WebServicePort	The port used to receive web messages.	None (*optional*)
SSLWebServicePort	True if SSL should be enforced on incoming web service connections.	True
ManagerSSLSocket	True if SSL should be used to connect to the DIVA Manager (*DIVA 7.6 or greater is required*).	False
AccessRulesFilename	The access rules file name. This file name is relative to the directory where the ManagerAdapter configuration file exists.	None (*if not configured, no access rules will be applied*)
WorkerThreads	The number of worker threads in DIVA Connect thread pools. Used for tuning large configurations. If you are unsure do not set this value.	50
LogDirectory	The folder where the log files will be produced.	The log/dicvaconnect/ManagerAdapter folder.
LogLevel	The detail level at which trace file logging takes place (***ERROR, WARN, INFO, DEBUG, TRACE***).	**INFO**
LogFileLifetimeHours	Number of hours to keep log files. Files older than this will be removed at the top of the hour.	Default: 50 Minimum: 1 Maximum: 1000

Table 4–8 ManagerDatabase Parameters

Parameter	Description	Default Value
Address	The IP Address of the DIVA Core Manager Database.	localhost
Port	The port used to access the database.	1521
User	The user name of the schema.	None (*required*)
Password	The password of the schema.	None (*required*)
DbSiteId	The Oracle SID	lib5
DbServiceName	The Oracle Service Name. Can be supplied instead of DbSiteId.	None (*optional*)

Table 4–8 (Cont.) *ManagerDatabase* Parameters

Parameter	Description	Default Value
SecureMode	A value of 1 enables SSL encryption of the database connection.	None (*optional*)
DbCreds	Contains the encrypted database credentials generated by DIVAConnectAdmin. This parameter cannot appear when Password is used.	None (*optional*)

An ObjFilter can be defined for each requesting sitename in the ManagerAdapter. The following table shows the valid parameters for the object filter:

Table 4–9 *ObjectFilter* Parameters

Parameter	Description	Default Value
RequestingSitename	The site name of the site requesting objects	None (*required*)
Category	Objects with the category provided will be synced to the requesting sitename. Multiple categories may appear.	None (*optional if CategoryPrefix appears*)
CategoryPrefix	The prefix will be prepended to every request received through the ManagerAdapter. Only objects with the category prefix will be synced to the requesting sitename.	None (*optional if Category appears*)

Configuring the DbSync Service

The DBSyncConfig.xml configuration file contains the configuration for the DbSync Service. Use the following steps along with the description of each parameter to configure the DbSync Service.

Insure that the DbSync service is running when using the ClientAdapter. If the DbSync service is not running, some requests that would ordinarily succeed may fail. For example, DIVA Connect restores of newly archived objects may fail, even though new DIVA Connect archive requests may succeed.

Modifying the DbSync Configuration File

The following tables describe the parameters that can appear in a DbSync configuration file. The file is in XML format. The default value column indicates the value that the parameter will have if the parameter is not specified in the configuration file. The column also indicates whether the parameter is required or optional.

To create a new configuration file:

1. Navigate to the DIVA Connect home directory (*where DIVA Connect is installed*).
2. Navigate to the Program/conf/divaconnect/templates folder, copy the DBSyncConfig.xml.ini to the parent directory, but leave out the .ini extension (*../DBSyncConfig.xml*).
3. Using the following Parameter Description Tables as a reference, edit the parameters in the DBSyncConfig.xml file to configure DbSync.

A sample DbSync configuration file appears in Appendix A.

Table 4–10 DbSync Top Level Parameters

Parameter	Description	Default Value
LocalSitename	The sitename of the local DIVA Connect site (*where DbSync is running*). The sitename listed must match what was configured in the DIVA Connect database and what is configured in both the ClientAdapterConfig.xml and ManagerAdapter.xml. This configuration enables DbSync to communicate with the ManagerAdapter.	None (*required*)
LogDirectory	The folder where the log files will be produced.	The log/divaconnect/Dbsync folder.
LogLevel	The detail level at which trace file logging takes place (***ERROR, WARN, INFO, DEBUG, TRACE***).	**INFO**
InternalPollingRateMillis	The base rate at which sites are polled for information. Change this parameter only if slow networks and (*or*) systems are encountered.	2000
WebServicePort	The port used to send management messages to DbSync.	None (*optional*)
SSLWebServicePort	True if SSL should be enforced on incoming web service connections.	true
BatchSize	The number of records to query at once when syncing site metadata	500
DoNotSyncFileList	**Advanced**: Do not synchronize filenames to the DIVA Connect database – this can greatly reduce sync times and disk usage.	False
AutoCopyRulesFilename	The name of the AutoCopy rules configuration file. **Note**: You must specify the AutoCopy rules filename here AND in the Client Adapter config if you wish to use AutoCopy!	None (*AutoCopy disabled*)
LogFileLifetimeHours	Number of hours to keep log files. Files older than this will be removed at the top of the hour.	Default: 50 Minimum: 1 Maximum: 1000

DivaManager Section

Inside of the DivaManagers tag, multiple DivaManager definitions can appear. Table 4–11 shows the parameters that can appear as part of a DivaManager section.

Table 4–11 DivaManagers Parameters for DbSync

Parameter	Description	Default Value
BaseUrl	Url of the service on the DIVA Manager platform to use for syncing. By default, this corresponds to the network address of the remote ManagerAdapter, qualified by the WebServicePort used by the ManagerAdapter.	None (*optional*)

Table 4–11 (Cont.) *DivaManagers* Parameters for DbSync

Parameter	Description	Default Value
Sitename	The official name of the site to sync object information from. The sitename listed must match what was configured in the DIVA Connect database and what is configured in both the ClientAdapterConfig.xml and ManagerAdapter.xml.	None (*required*)

DIVA Connect Database

Configure the DIVA Connect Database parameters as shown in Table 4–12.

Table 4–12 *DIVAConnectDatabase* Parameters

Parameter	Description	Default Value
Address	The IP Address of the database.	localhost
Port	The port used to access the database.	1521
User	The user name of the schema.	None (*required*)
Password	The password of the schema.	None (*required*)
DbSiteId	The Oracle SID (*Site Identifier*).	lib5
DbServiceName	The Oracle Service Name. Can be supplied instead of DbSiteId.	None (*optional*)
SecureMode	A value of 1 enables SSL encryption of the database connection.	None (*optional*)
DbCreds	Contains the encrypted database credentials generated by DIVAConnectAdmin. This parameter cannot appear when Password is used.	None (*optional*)

Access Rules

In DIVA Connect, you use Access Rules to control access to DIVA Connect operations, sites, and resources. Access rules can be run in three ways:

- On DIVA Connect requests within the ClientAdapter service (*in MultiDiva Mode or sent via DIVA Connect UI*)
- On DIVA Core requests within the ManagerAdapter service
- On DIVA Core requests that enter the ClientAdapter through a Direct Mode port

To run Access Rules, you must define the AccessRulesFilename parameter in the ClientAdapter and (*or*) ManagerAdapter configuration file. You must supply the file name without the path - DIVA Connect assumes that the file will be located in the same directory as the ClientAdapter configuration file.

Methods of Running Access Rules

Access rulesets defined in the ClientAdapter (*MultiDiva Mode*) enforce access control on DIVA Connect requests (*received locally*). Access rulesets defined in the ManagerAdapter configuration enforce access control on DIVA Core requests (*submitted to satisfy a DIVA Connect request*). The rules in the ManagerAdapter implement policies for the site. The rules in the ClientAdapter implement policies for the system as a whole.

Access rulesets defined in the ClientAdapter (*Direct Mode*) enforce access control on either DIVA Connect requests or DIVA Core requests, depending on whether the remote system is another DIVA Connect instance, or a DIVA Core system. In this mode, additional operation types are available for use in rulesets. These operations correspond to DIVA Core-specific requests, and are detailed below.

Archive Example

Let's quickly jump into an example to get more familiar with the rules themselves. The following rule allows Archive operations for users connecting as *admin* or as *operator* from either Server *DATA_EXP_PDAT1* or *VID_FTP_3*, and, archiving to a media of either *HDFeatures* or *spm* (*you will see that the order of the attributes is not significant*), and being named with a category containing the word *POST*.

```
<Include>
   <Operation>Archive</Operation>
   <Username>admin</Username>
   <Username>operator</Username>
   <ReqMedia>spm</ReqMedia>
   <ReqObjectCategory>*POST*</ReqObjectCategory>
   <ReqSourceDest>DATA_EXP_PDAT1</ReqSourceDest>
   <ReqSourceDest>VID_FTP_3</ReqSourceDest>
   <ReqMedia>HDFeatures</ReqMedia>
</Include>
```

Copy Example

In the following two rules, guest users from the *GUI Workflow Profile* are not allowed to copy from diva2 to diva3, or vice versa.

```
<Exclude>
   <WorkflowProfile>GUI</WorkflowProfile>
   <Username>guest</Username>
   <Operation>Copy</Operation>
   <SourceSitename>diva2</SourceSitename>
   <TargetSitename>diva3</TargetSitename>
</Exclude>
<Exclude>
   <Username>guest</Username>
   <WorkflowProfile>GUI</WorkflowProfile>
   <Operation>Copy</Operation>
   <SourceSitename>diva3</SourceSitename>
   <TargetSitename>diva2</TargetSitename>
</Exclude>
```

You used two rules here because you didn't want to explicitly restrict copy operations that occur within the same site. For instance, someone at site diva2 may want to copy an object (*using DIVA Connect*) to a new tape - in this case the source and target sitename are both *diva2*! If you had a single rule that contained all of the SourceSitename and TargetSitename attributes, you would exclude copies from *diva2* to *diva2* and from *diva3* to *diva3*!

You are still not done. The copy will not succeed unless you have at least one Include rule that matches.

```
<Include>
   <Operation>Copy</Operation>
   <WorkflowProfile>GUI</WorkflowProfile>
   <Username>guest</Username>
   <Operation>ApiConnect</Operation>
</Include>
```

In this case, a very general Include rule gives you what you want, the ability to copy everywhere except for *diva2* to *diva3* and vice versa. In fact, you didn't really need an Exclude rule at all. Sometimes, however, things are easier with Exclude rules. Keep in mind that if an Exclude rule matches an operation, that operation will be denied, even if one or more Include rules match.

Include and Exclude Rules

To summarize, there are two types of rules, Include and Exclude. Access is denied for all requests unless at least one Include rule matches the operation about to be performed. However, if any Exclude rule matches, the operation is automatically rejected, regardless of any Include rules that match.

Attribute Types

On requests such as Delete, Copy, Restore, Partial File Restore, Cancel, and Archive, DIVA Connect runs the entire set of Access Rules set to see if the operation is permitted. It examines variables such as:

- **Originator attributes** — the Workflow Profile of the connection, the user name that sent the message, the IP address of the originator.
- **Request attributes** — the Server, the source/target sitenames, the requested media, comments, and so on. These are derived from the request itself. Many of these are prefixed with Req.
- **Object attributes** - the media(*s*) the object is stored on, the storage plan, the object size, and so on. These are derived from the object being processed by an operation. Many of these are prefixed with Obj.

The following rule combines all three attribute types. It allows a user *diva* to perform a site delete on New York only when the object exists on Los Angeles.

```
<Include>
   <Username>diva</Username>
   <Operation>Delete</Operation>
   <SubType>SiteDelete</SubType>
   <TargetSitename>NewYork</TargetSitename>
   <ObjOnSite>LosAngeles</ObjOnSite>
</Include>
```

Rules for DIVA Connect Requests (*ClientAdapter*)

DIVA Connect requests are generated when requests are received in MultiDiva Mode. Access rules can be created against these DIVA Connect operations. The following are details of some attributes specific to DIVA Connect requests.

Connect Operations

The ApiConnect and WebConnect operations are special operations that must be included to establish a connection to the ClientAdapter.

- **ApiConnect** — This operation governs the ability to connect to the ClientAdapter through an API client socket connection. This is required for DIVA API connections.
- **WebConnect** — This operation governs the ability for applications to connect through web connections (*DIVAConnectUI and DIVAConnectAdmin*). This is required for DIVAConnectUI connections.

When these operations are matched against your rules, keep in mind that only originator attributes will be present to match against. For example, TargetSitename would not be matched when access rules are run during the ApiConnect operation, because the attribute is simply not present when a client connects.

SubType (*for Delete*)

The delete operation has a SubType field which represents a subtype of the operation. You can include the SubType field in rules along with the Delete operation, supplying multiple SubType parameters if you want. The values for Delete SubType are:

- **GlobalDelete** — Matches if the delete operation being performed is a global delete of an object on all sites. This also matches a site delete operation that just happens to be deleting all remaining objects in DIVA Connect.
- **SiteDelete** — Matches if the delete operation is a delete of all instances on a particular site (*the name of the site can be matched in rules using the TargetSitename parameter*). In addition, an operation will have this SubType if the requester is deleting a single instance, but that is the last instance of the object on that site.
- **InstanceDelete** — This delete operation deletes a single instance on a site, and there are other instances on the site.

Specifying this parameter in rules is useful for enforcing the scope of the permitted delete operations.

Rules for DIVA Core Requests (*ManagerAdapter*)

DIVA Connect allows access rules to be run on DIVA Core requests as well. Access rules defined in the ManagerAdapter specify which DIVA Core operations (*sent to satisfy DIVA Connect requests*) are permitted. Only the operations valid for DIVA Connect requests can be specified in the rulesets. In ManagerAdapter rulesets, the WorkflowProfile, TargetSitename, and SubType attributes are invalid.

Similar to the ClientAdapter rulesets, the WebConnect Operation must be granted for connections to be established to the ManagerAdapter. This also extends to DbSync operations. The SourceSitename attribute corresponds to the specific site that is making a request. The ApiConnect operation is not available in ManagerAdapter Access Rules.

Rule Matching

Matching a rule involves comparing attributes in the rule against the actual values in each request, in a matching object, or request originator. Wildcards are allowed (*use an asterisk (*) for wildcards*). Case is insensitive for the tag names, but most values are case sensitive. The Operation tag is required in each rule. You can provide an Operation tag containing an asterisk (*) to indicate that the rule applies for all operations. However, you should be careful when doing this because not all attributes are valid for all operations.

Distinct attributes in a rule (*an Include or Exclude rule*) are joined together using a logical AND in the matching process. However, a single attribute specified more than one time in a rule causes those attributes to be joined together using a logical OR in a single expression.

When matching a requested operation against an access rule, DIVA Connect determines if the attribute in the rule is applicable for the operation being performed. If not, the attribute is not used in the comparison.

Rulesets and Defaults

Rules can be grouped into Rulesets. Each Include-Exclude rule must be contained inside of Ruleset tags. Each incoming DIVA Connect operation is matched against all rulesets. Rulesets are useful because they can have attributes that serve as defaults for all rules contained inside. In the matching process, each Ruleset default attribute is folded in to each child rule - as if the attribute were specified directly within the rule. It is common to use Workflow Profile as a default attribute for Rulesets, as Workflow Profile is populated on every requested DIVA Connect operation.

The following attributes can be defaulted within a ruleset:

- User Name
- Net Address
- WorkflowProfile (*in ClientAdapter rules*)

Another Example

The following example allows users connected with a WorkflowProfile of **GUI** to view requests and assets, and to perform deletes of individual instances on site *diva1*. Remember that a SubType of **InstanceDelete** will deny any delete that would remove the last instance of a particular object on a site.

The second part of the example prohibits all connections, either web or API connections, that originate from the subnet 172.53. It will do this for any operation, regardless of its Workflow Profile.

```
<Ruleset>
  <WorkflowProfile>GUI</WorkflowProfile>
  <Include>
    <Operation>WebConnect</Operation>
    <Operation>Delete</Operation>
    <SubType>InstanceDelete</SubType>
    <TargetSitename>diva1</TargetSitename>
  </Include>
</Ruleset>
<Ruleset>
  <NetAddress>172.53*</NetAddress>
  <Exclude>
    <Operation>ApiConnect</Operation>
    <Operation>WebConnect</Operation>
  </Exclude>
</Ruleset>
```

Ruleset Parameters

Table 4–13 shows the parameters that can appear under the Ruleset section.

Table 4–13 Ruleset Parameters

Parameter	Description	Default Value
WorkflowProfile	This attribute, defined in the ClientAdapter configuration, is the name of a workgroup or set of applications that access DIVA Connect. This attribute is made part of every rule in the ruleset.	None (*optional*)

Table 4–13 (Cont.) Ruleset Parameters

Parameter	Description	Default Value
Username	The user name passed in the API or specified in the web request. This attribute is made part of every rule in the ruleset.	None (*optional*)
NetAddress	The network address (*IP or internet name*) of the remote connection. This may be the address of a gateway or router and not the address of the originator. This attribute is made part of every rule in the ruleset.	None (*optional*)
Exclude	A rule that denies permission if it matches the DIVA Connect operation.	None (*optional*)
Include	A rule that allows permission if it matches the DIVA Connect operation.	None (*optional*)

Include/Exclude Rule Parameters

You can specify rule attributes that appear in requests. For example, ReqMedia will match the media and (*or*) storage plan specified in a request (*operation*).

Similarly, you can specify rule attributes that match against the archived object that the request is processing. For example, if a certain object is specified in a delete operation, ObjHasMedia will match any media that is currently part of that archived object regardless of media passed in the request.

Table 4–14 shows the parameters that can appear in an Include or Exclude rule section.

Table 4–14 Rule Operational Parameters (Include or Exclude)

Parameter	Description	Default Value
Operation	The name of the DIVA Connect operation to match: - Archive - Copy - CopyToNew - Restore - PartialRestore - TranscodeArchived - Delete - DeleteFile - Cancel - ApiConnect - WebConnect - ChangeConfig There are additional operations available for DIVA Core operations in Direct Mode: - InsertTape - EjectTape - AssociativeCopy - TransferFiles - ServerDelete - ChangePriority	None (*optional*)
Username	The user name of the connected API user and (*or*) service user.	None (*optional*)
WorkflowProfile	The ClientAdapter profile name.	None (*optional*)
NetAddress	The IP address of the API application and (*or*) user.	None (*optional*)
SourceSitename	The source sitename of the operation. Some operations do not have a source sitename (*for example, Archive has a Server as the source*). If the request is retried on another site, this value will change, and the entire rule will be reevaluated. When this attribute appears in ManagerAdapter rules, this matches the site that submitted the request.	None (*optional*)
TargetSitename	The target sitename of the operation. Some operations do not have a target sitename (*for example, Restore has a Server as the target, not a site*). If the request is retried on another site, this value will change, and the entire rule will be reevaluated.	None (*optional*)
SubType	Multi DIVA Mode operation type (*Delete command has GlobalDelete, SiteDelete, and InstanceDelete*).	None (*optional*)

Table 4–15 Rule Request Parameters (Include or Exclude)

Parameter	Description	Default Value
ReqObjectName	The name of the object that is being processed.	None (*optional*)
ReqObjectCategory	The category of the object (*in DIVA Core this is part of the formal name of the object*).	None (*optional*)

Table 4–15 (Cont.) Rule Request Parameters (Include or Exclude)

Parameter	Description	Default Value
ReqSourceDest	The Server specified in the request.	None (*optional*)
ReqComments	The comments field in the request.	None (*optional*)
ReqMedia	The media that was requested as part of the operation/request (*sitename should not be appended*). Keep in mind that *Storage Plan* can be passed as the requested media.	None (*optional*)
ReqOptions	The options field in the request.	None (*optional*)
ReqPathRootPat	The file path root in the request. Asterisk (*) matches up to the next forward slash (/), double-asterisk (**) matches all characters.	None (*optional*)

Table 4–16 Rule Object Parameters (Include or Exclude)

Parameter	Description	Default Value
ObjOnSite	Matches if the object exists on the specified site.	None (*optional*)
ObjNotOnSite	Matches if the object is NOT on the specified site.	None (*optional*)
ObjHasMedia	Matches any media on any site.	None (*optional*)
ObjHasStoragePlan	Matches storage plan on any site (*ObjStoragePlanSite can restrict to one site*).	None (*optional*)
ObjStoragePlanSite	Restricts specified storage plan to a specific site.	None (*optional*)
ObjHasSizeGbLessThan	Matches if total object size in fractional GBs is less than the specified value.	None (*optional*)
ObjHasSizeGbGreaterThan	Matches if total object size in fractional GBs is greater than the specified value.	None (*optional*)
ObjSourceDest	Matches the original Server stored in the asset database.	None (*optional*)
ObjPathRootPat	The file path root in the request. asterisk (*) matches up to the next forward slash (/), double-asterisk (**) matches all characters.	None (*optional*)

AutoCopy

AutoCopy is a feature that can automatically copy or restore assets after they are archived. AutoCopy rules are evaluated when new content is detected at a site. The rules can trigger one or more restore or copy tasks that are executed according to a schedule. AutoCopy rules can also be evaluated in batch mode against all assets.

- AutoCopy Rulesets
- AutoCopy Queue
- Scheduling AutoCopy Tasks
- Configuring AutoCopy

AutoCopy Rulesets

AutoCopy rulesets control which new objects will be restored or copied, and with what settings. You can define multiple rulesets in the AutoCopy configuration file (*see "Creating a New AutoCopy Configuration File"* on page 4-31).

Each ruleset contains the following:

- A set of Include/Exclude rules — These act as "if" statements that describe which assets that should be restored or copied, and which should not (*for a list of valid parameters, see " Include/Exclude Rule Parameters"* on page 4-32).
- A single Action section — This acts as a "then" statement that determines what site to copy to, or what Server to restore to (*for a list of valid parameters, see " Action Section Parameters"* on page 4-33).

A new asset matches a ruleset when at least one Include rule matches and no **Exclude** rule matches. DIVA Connect uses the parameters in the ruleset Action section to generate either a copy or restore task. AutoCopy rules can create tasks for assets originating from any DIVA Connect site (*not just the local site*).

The format and behavior of AutoCopy Rules loosely match the behavior of Access Rules (*see "Access Rules"* on page 4-19). Like Access Rules, repeating the same attribute within a rule creates an “OR” condition for the attribute. However, unlike Access Rules, a newly archived object can match multiple rulesets, and consequently create multiple copy or restore tasks.

AutoCopy rules are evaluated:

1. When a new asset it is archived at a particular site, or
2. On all existing assets, when you select the “Run Rules on Existing Assets” option in the DIVAConnectAdmin utility (*see "Managing the AutoCopy Queue with DIVAConnectAdmin"* on page 4-29).

Sample AutoCopy Rulesets

Example 4–1 Basic AutoCopy Ruleset

This ruleset will copy object assets from NewYork that have a storage plan of "SP_MEDIA1" and are less than 300GB in total size. If a match occurs, DIVA Connect will create a copy in the LosAngeles site with a media of "LTArch_DVT_M2" and a priority of 50.

```
<Ruleset>
  <Include>
    <SourceSitename>NewYork</SourceSitename>
    <ObjHasStoragePlan>SP_MEDIA1</ObjHasStoragePlan>
    <ObjHasSizeGbLessThan>300</ObjHasSizeGbLessThan>
  </Include>
  <Action>
    <Operation>Copy</Operation>
          <Sitename>LosAngeles</Sitename>
    <Media>LTArch_DVT_M2</Media>
    <Priority>50</Priority>
  </Action>
</Ruleset>
```

Example 4–2 Complex AutoComplex Ruleset

The following ruleset will copy assets from Dallas that have a category starting with "POST" (*but not having an object name of AV_49082_DVT*). It will also copy objects archived in NewYork that have the following:

1. An original Server of either "Interplay_001" or "Interplay_002", and
2. An object name starting with "AV_" and ending with "_DVT", but
3. Not a name of "AV_49082_DVT" and a category of "POST".

```
<Ruleset>
  <Include>
    <SourceSitename>Dallas</SourceSitename>
    <ObjectCategory>POST*</ObjectCategory>
  </Include>
  <Include>
    <SourceSitename>NewYork</SourceSitename>
    <ObjSourceDest>Interplay_001</ObjSourceDest>
    <ObjSourceDest>Interplay_002</ObjSourceDest>
    <ObjectName>AV_*_DVT</ObjectName>
  </Include>
   <Exclude>
         <ObjectName>AV_49082_DVT</ObjectName>
         <ObjectCategory>POST</ObjectCategory>
      </Exclude>
     <Action>
    <Sitename>LosAngeles</Sitename>
    <Media>LTArch_DVT_M1</Media>
    <Priority>35</Priority>
    <DelayMins>1440</DelayMins>
  </Action>
</Ruleset>
```

In the case of a match, DIVA Connect will copy the new object (*from either NewYork or Dallas*) to the LosAngeles site one day after the copy task is created.

AutoCopy Queue

The AutoCopy queue is a list of copy or restore tasks waiting to be executed. The MaxRequests parameter limits the number of AutoCopy requests executing at one time. By default, DIVA Connect uses a combination of task priority and the last retry date to determine which tasks to schedule next. For additional scheduling options, see "Scheduling AutoCopy Tasks" on page 4-30.

What happens on a copy if the object is already present on the target site? If an AutoCopy task's object name and category matches an asset on the target site, and is stored on the target media, DIVA Connect will not create a copy, and will mark the task as succeeded.

If the object already exists on the target site, but is not stored in the target media, DIVA Connect, again by default, performs a local copy to group operation at a local site. However, if the AutoCopy parameter AllowLocalCopy is set to false, DIVA Connect will not generate a copy request, and instead will mark the task as succeeded.

Finally, if the -replaceTarget parameter is selected (*either in AutoCopy rules or the options field of an API request*), AutoCopy will delete the object from the target site before copying it back to the target site.

When are tasks removed from the queue? After an AutoCopy task succeeds in copying or restoring an object, the task is complete, and DIVA Connect removes the task from the queue. For example, if an object is copied to a site, and later the object is deleted from the site, there will be no action taken to re-copy the asset to the target site. If an asset is deleted from all sites, the corresponding entries in the AutoCopy queue are also deleted.

What happens when a request fails? If Autocopy schedules a restore or copy request and the request fails, AutoCopy will create new requests at regular intervals until one succeeds. You can define RetryTotalLimit and RetryDelayMins to control the retry attempts.

This retry is distinct from the retry options available in the Workflow Profile, which can retry some of the steps within a request without terminating the request. For example, when issues occur on the target DIVA site during an intersite copy, the archive step of the transfer can be retried without repeating the download to the target site. In contrast, the AutoCopy retry creates a brand-new copy request with a distinct request ID. You can combine the two types of retry to provide an effective and reliable method of transfer.

Managing the AutoCopy Queue with DIVA Connect UI

- **Viewing AutoCopy Assets** — To view AutoCopy assets, set the “Has AutoCopy Task” filter to “True” in the asset search panel (*see* "Searching Assets" on page 5-3). This returns all assets that have an AutoCopy task either waiting to be scheduled or running. On the asset detail panel, you can see the tasks for an asset, as well as any running requests. You can view the details of a running request on the DIVA Connect UI requests screen.
- **Canceling AutoCopy Tasks** — To cancel tasks, select the assets that have tasks you want to remove, and then select **Cancel Tasks** from the Action menu. By default, this cancels all tasks associated with the assets. A dialog box is displayed allowing deletion of all, or some, tasks associated with the assets. Tasks originating from a particular site, or tasks having a specific target media can be canceled. DIVA Connect cancels a task by removing the task from the AutoCopy queue. Canceling has no effect on any copy requests that may be currently running.

Managing the AutoCopy Queue with DIVAConnectAdmin

- **Pause/Unpause and Clear AutoCopy Tasks** — Using DIVAConnectAdmin, you can pause and unpause the scheduling of new requests. Pausing or unpausing will not effect requests that AutoCopy has already spawned. You can also use DIVAConnectAdmin to clear all tasks in the AutoCopy queue. See " DIVAConnectAdmin Options" on page 6-1.
- **Evaluate AutoCopy Rules Against Existing Assets** — You can select “Run Rules on Existing Assets” option in the DIVAConnectAdmin utility to generate copy tasks for any existing asset that matches the AutoCopy rulesets. See " DIVAConnectAdmin Options" on page 6-1. Starting with DIVA Connect 3.1 this option now accepts an optional file name. If a file is entered, then the rules rulesets from this file will be evaluated. If no file is specified, the AutoCopy rulesets specified in the ClientAdapterConfig will be used.

An AutoCopyRules file used for this feature uses only the <Ruleset> section. The top-level parameters are not used and must be specified in the file that is referenced in the ClientAdapterConfig and DbSyncConfig files.

Caution: Running AutoCopy rules on existing assets can result in thousands of copy tasks added to the AutoCopy queue.

Routing Copy or Restore Requests to the AutoCopy Queue Using DIVA API

Restore or Copy requests sent to DIVA Connect (*running in MultiDiva mode*) via the DIVA API normally execute immediately. However, there is an option to route requests to the AutoCopy queue for scheduling. The request becomes a task in the queue, allowing an application to schedule thousands of restores or copies at once, just by using the DIVA API. These requests may take weeks to complete.

To route a request to AutoCopy, provide the -autocopy flag in the options field of a Copy or Restore command sent via the DIVA API. *Partial Restore is currently not supported*. DIVA Connect will use the parameters in the request to generate the task. The -autodelaymins parameter will delay the activation of the task for a specific number of minutes.

Routing Copy or Restore Requests to the AutoCopy Queue Using DIVA Connect UI

You can optionally route DIVA Connect UI copy or restore requests to the AutoCopy queue. This facilitates the scheduling of thousands of copies at one time from a remote application.

Scheduling AutoCopy Tasks

When tasks make it into the queue, AutoCopy can then schedule them for execution. The following are a few features that can be used to schedule tasks that are in the AutoCopy queue. These features can make the best use of available tape, disk, and cloud resources.

Schedule by Tape

When copying or restoring large numbers of tape-based assets, tape library mounts, dismounts, and positioning can become significant. For items in the queue, the Schedule by Tape feature will group together assets that exist on the same tape, and use these groupings to schedule copies. This can help minimize the total time to restore tape-based assets.

To enable the feature, set the NumTapesToSchedule parameter to a value greater than zero.

The NumTapesToSchedule parameter indicates how many tapes to schedule concurrently (*it does not indicate the number of tasks or assets to schedule*). This value should be set to the maximum number of tape drives to dedicate to copying or restoring. DIVA Connect selects a list of tapes to copy, picks assets from each tape, and then schedules them in a round-robin fashion, until it reaches the MaxRequests limit. The outbound request priority is set using the ScheduleByTapePriority parameter. Using Schedule By Tape will always override any priority specified in AutoCopy rules, as priority affects request ordering in DIVA.

When executing schedule by tape, the Copy tasks normally execute a restore request at the source site, followed by an archive at the target site. In the event the source site has disk instances as well as the selected tape instance, DIVA Core may select the disk instance. This may be ok, but it also may lead to the tape getting dismounted only to be mounted again for the next asset. To avoid this, and reduce the number of mounts and dismounts, starting with version 3.1 the RestoreInstanceOnCopyTape parameter may be used to send a restore instance request to the source site. The tape instance will then be selected even if a disk instance exists.

NumTapesToSchedule must be set to put the Auto Copy in ***Tape Priority*** and enable the ***Restore Instance*** feature.

Additionally, you can set up Schedule by Tape to run only during off-peak hours, if desired. See below.

Off-Peak Time Window

Using AutoCopy, you can define a daily “off-peak” time window during which DIVA Connect can execute more requests at one time (*or less requests, if desired*).

The daily start time (*OffPeakStartTime*) and stop time (*OffPeakStopTime*) define the off-peak window. If either values are omitted (*or they are the same*), no off-peak period will occur.

By configuring OffPeakNumTapesToSchedule you can run the Schedule by Tape feature during off-peak hours. See above for more information.

Configuring AutoCopy

The AutoCopy configuration file (*AutoCopyRules.xml*) contains:

- AutoCopy Configuration Parameters
- AutoCopy Rulesets

If you modify the AutoCopy configuration file while the system is running, select the “Reload Workflow Profile / Rules” option in DIVAConnectAdmin to inform DIVA Connect of your changes without restarting services (*see* " DIVAConnectAdmin Options" on page 6-1).

Creating a New AutoCopy Configuration File

1. Navigate to the DIVA Connect home directory (*where DIVA Connect is installed*).
2. Navigate to the Program/conf/divaconnect/templates folder.
3. Copy the AutoCopyRules.xml.ini to the parent directory, but leave out the .ini extension (*../AutoCopyRules.xml*).
4. Using the tables below as a reference, edit the parameters in the AutoCopyRules.xml file to configure the AutoCopy feature.
5. To activate AutoCopy, you must add a reference to the AutoCopy config file in both the ClientAdapter and DbSync configuration files. The AutoCopyRulesFilename parameter indicates the name of the AutoCopy file. The services assume the AutoCopy file is in the same directory as the other service configuration files.

Caution: Be careful when adding the AutoCopy file to the ClientAdapter and DBSync configuration files. Errors in the AutoCopy file may prevent the services from starting, in the same way as errors in the ClientAdaptor or DBSync configuration files.

Creating a New BatchAutoCopy Configuration File

Create the BatchAutoCopy the same as the AutoCopy Configuration file, using items 1 - 4 i the previous section, except in item 3, use BatchAutoCopyRules.xml.ini as the template. Note that the top level configuration items are not included in this file. These parameters continue to be specified by the AutoCopy rules defined in ClientAdapterConfig and DbSyncConfig. This file may be specified when using the *Run Rules on Existing Assets* option in the *Manage AutoCopy Menu* of the Admin tool.

AutoCopy Configuration Parameters

The following table contains the configuration file top-level parameters used to customize scheduling behavior. The default value in the Data Type/Default column indicates the value that the parameter will have if it is not specified in the configuration file.

Table 4–17 Configuration Parameters

Parameter	Description	Data Type/Default
MaxRequests	Maximum simultaneous AutoCopy requests to schedule at one time (*during normal processing*)	Integer (*default: 10*)
NumTapesToSchedule	Turns on Schedule By Tape if greater than zero; Schedule this number of tapes at a time	Integer (*default: 0=disable scheduling by tape*)

Table 4–17 (Cont.) Configuration Parameters

Parameter	Description	Data Type/Default
AllowLocalCopy	Allows local copies using AutoCopy. If true, and the asset already exists at the target site, a local copy to the target media will be scheduled (*provided the object is not already stored on that media*).	Boolean (*default: true*)
DefaultWorkflowProfile	The DIVA Connect Workflow Profile to use when submitting copy requests	String (*default: default*)
ScheduleByTapePriority	Copy request priority to use when Schedule by Tape is enabled. Schedule by Tape overrides the priority specified in AutoCopy rules.	Integer (*default: 40*)
RestoreInstanceOnCopyTape	If true the restore request of a copy (*tape priority*) will use restoreInstance.	Boolean (*default: false*)
RetryTotalLimit	Maximum number of retries for an AutoCopy task	Integer (*default: 3000*)
RetryDelayMins	After an AutoCopy request fails, the minimum delay before another attempt	Integer (*default: 15 minutes*)
DiskRequestReserve	How many requests of the total to be reserved for disk-based assets, when scheduling by tape	Integer (*default: 1*)
OffPeakStartTime	Time to start the off-peak period	Time of day (*HH:MM, local 24hr*) (*default: none: no off-peak period*)
OffPeakEndTime	Time to end the off-peak period	Time of day (*HH:MM, local 24hr*) (*default: none: no off-peak period*)
OffPeakMaxRequests	Maximum simultaneous AutoCopy requests to schedule during the off-peak period	Integer (*default: 40*)
OffPeakNumTapesToSchedule	Number of tapes to schedule during off-peak hours	Integer (*default: 0=no scheduling by tape*)
ConsecTapeErrorThreshold	Number of errors before abandoning a tape (*when scheduling by tape*)	Integer (*default: none*)

Include/Exclude Rule Parameters

AutoCopy rulesets determine which assets should replicate to other sites. Each ruleset contains if portions (*the Include sections of each ruleset*), and a then portion (*the Action section of each ruleset*). See AutoCopy Rules above for examples of rules and actions.

Table 4–18 Include/Exclude Rule Parameters

Parameter	Description	Data Type/Default
SourceSitename	Matches the sitename where the asset was archived	String

Table 4–18 (Cont.) Include/Exclude Rule Parameters

Parameter	Description	Data Type/Default
ArchivedMoreThanDays	Matches if asset was archived more than this many days ago	Integer (*number of days*)
ArchivedLessThanDays	Matches if asset was archived less than this many days ago; useful if a DIVA Connect database resync occurs	Integer (*number of days*)
ObjectName	The name of the asset	String
ObjectCategory	The category of the asset (*part of the formal name of the asset*)	String
ObjComments	The comments field in the new asset	String
ObjHasMedia	Matches the media to which the asset has been archived	String
ObjSourceDest	Matches the Server where the object was archived from	String
ObjHasStoragePlan	Matches storage plan of new asset	String
ObjHasSizeGbGreaterThan	Matches if the asset size is greater than this number (*in fractional GBs*)	Decimal (*in GBs*)
ObjHasSizeGbLessThan	Matches if the asset size is less than this number (*in fractional GBs*)	Decimal (*in GBs*)
ObjPathRootPat	Matches the original file path root of the archived object. Glob matching (** matches up to next /, ** matches all*)	String

Table 4–19 Action Section Parameters

Parameter	Description	Data Type/Default
Sitename	The name of the DIVA Connect site to copy to	String (*required*)
Media	The target archive media to copy to	String (*required*)
Priority	The priority of the request	Integer (*1-100; default: 50*)
DelayMins	The time to wait in minutes before scheduling the task	Integer (*default: 0*)
Operation	The name of the operation (*Copy or Restore*)	String (*default: Copy*)
ReplaceTarget	True if the copy should replace that already exists on the site	Boolean (*default: false*)
SourceDest	The Server to restore to	String (*1-96*)
FilePathRoot	The FilePathRoot to use in the restore	String
Options	The options to use for the copy/restore	String
CreateRestoreFolder	True if a uniquely named top-level folder should be created for each object	Boolean (*default: false*)

Table 4–19 (Cont.) Action Section Parameters

Parameter	Description	Data Type/Default
WorkflowProfile	The DIVA Connect Workflow Profile to use when submitting copy requests.	String (*default: DefaultWorkflowProfile*)

Adding Script Variables to a Configuration File

DIVA Connect allows variable substitution in configuration files to make configuring multiple script files easier. When the pattern *${variable_name}* is encountered in an XML Value (*variables are not valid in Tag Names*), the value of the variable is substituted when the DIVA Connect reads the script. The value can be taken from an environment variable or assigned directly within the script.

If the Variable Name in the script is named the same as an Environment Variable, the value of that variable will be substituted in the script. Alternatively, you can define variable values at the top of each script using the Variable tag. The syntax is: <Variable name="LocalSitename" value="diva1"/>. When DIVA Connect reads a configuration script, it checks to see if any variables that are used within the script have been set at the top. If not, it then looks for an Environment Variable with the same name as the variable.

For ease of use, a script file named divaconnectEnv.conf has been created within the DIVA Connect home directory, in the Program\conf\divaconnect\wrapper subfolder. Variables can be centrally defined in this configuration file and will be available to all of the DIVA Connect services. If the values are changed and the service is restarted, the changes will be reread. Refer to the divaconnectEnv.conf file for sample variables.

5

Using the DIVA Connect User Interface

Prerequisites

To use the DIVAConnectUI application:

- You will need to have the ClientAdapter configured and running on the server side.
- If you want to perform intersite copies within DIVAConnectUI, ensure that the ClientAdapter Site-To-Site mappings are configured.
- To view the most up-to-date asset information, you will want to configure and run the DbSync service.

> **Note:** DIVA Connect 2.x does not supports the DIVA Core Control GUI — use the DIVAConnectUI instead. DIVA Connect 2.x does not forward or support Control GUI messages in either MultiDiva or Direct Mode.

Starting the DIVAConnectUI

To start the DIVA Connect UI application in Windows, click the provided shortcut link. This link is located in the DIVA Connect home directory, in the Program/divaconnect/bin subfolder. This shortcut can be copied to the Windows Desktop or Windows Start menu.

On Linux, open a terminal window and call the DIVAConnectUI.sh script to start the DIVAConnectUI.

Refer to Chapter 3 for details on how to install the DIVAConnectUI.

Connecting to DIVA Connect

After starting the DIVA Connect application:

1. Click either the **Requests** or **Assets** tab to open a prompt for the DIVA Connect server URL.
2. Enter the URL of the DIVA Connect server, and then click **OK**.

   By default, the URL corresponds to the host name of the ClientAdapter, followed by a colon, followed by the ClientAdapter WebServicePort. The value that you enter will be persisted in the DIVAConnectUI properties file.

   For example: https://127.0.0.1:9801
3. If userlogin=true in divanetui.properties, a basic authentication dialog will open to accept a username and password. See *Using the DIVAConnectAdmin Utility,* Table 6–1, *Task: Add a User*, for information on adding users to the database. A username and password are required only if UserAuthEnabled at the TOP_LEVEL of ClientAdapterConfig is true.

If a connection error occurs, a message will appear in the lower right corner of the screen. Clicking the **Change URL** button will prompt for a new URL.

Viewing Assets

To display a list of assets, click the **Assets** tab. The first 100 assets will be listed in the left pane. The asset name will be shown in blue, followed by the object category.

By default, assets are presented in Object Name order. Select an asset to view details displayed on the right pane.

Asset Details Panel

The Asset Details panel is divided into three sections: the Global Object Parameters at the top, the Sites Table in the middle, and the Asset Filenames table at the bottom. The following are descriptions of some parameters and sections on the Asset Details panel:

- **Object Parameters** — The parameters at the top of the Details panel include the total size of the object, the archive date, the number of copies (*instances*), and comments.
- **Last Verified Date** — This represents the last time that the object was checksum verified on tape, at any site. If no tape instances exist, only then will it return the last time the object was verified on disk. If the object was never verified on archived media, the field is blank.
- **Alerts** — Any object alerts will be displayed directly underneath the Comments field. For instance, if DIVA Connect detects that two sites have objects with the same name, but different content, an alert will appear.
- **Active Tasks & Requests** — On the Asset Details panel, the text above the Sites Table indicates any DIVA Connect tasks or requests currently active on the selected object. Each task will have the hours that the task has been active, or, the number of hours until the task becomes active. DIVA Connect requests will have the request type and id displayed. It is useful to view this field before issuing copies or deletes. This value is refreshed only when you press the **Search** button (*see* Searching Assets).
- **Sites Table** — Each site that contains instances of the object is displayed in the site table. Each site can contain one or more object instances. Objects for each site have a Storage Plan and initial archive dates (*which can be used in the Search panel*).

  A site icon resembling a house indicates that the site is the local site. A purple site icon indicates that the site's assets are display-only, and requests cannot be run on them. A site icon resembling a globe indicates that the site is not local.

- **Asset Filenames Table** — The file names that comprise the archived asset appear below the site table. The file names are organized into pages, and ordered according to the order that they were archived. The file names for complex archived objects cannot be displayed in this panel.

Searching Assets

DIVAConnectUI contains powerful object search capabilities. Click the **Search** button on the left to open the Asset Search panel.

The Asset Search panel allows many ways to search for assets. Some parameters are selectable, meaning that you can select both an attribute and value. Each value entered poses an additional restriction on the assets returned. Wildcards are permitted in all search fields except the date and time fields.

The following is a list of some parameters used for searching assets:

- **Archive Date Search** — DIVA Connect allows the user to search on the date that an object was archived to a DIVA site. A start and end date and time range can be specified that narrows the search to objects that have an archive date in range. Click the calendar icon to select a date, and then modify the time value in the text field to the desired time. An object may have multiple archive dates (*one for each site*), and this query searches for any that exist within the range.
- **Object Name / Category** — An asset name is composed of two fields, Object Name and Object Category. The Category is the second half of the formal name of an object. Additional object names and/or categories can be queried using user-selectable query terms (*see Other Search Parameters below*).
- **Site Search** — You can search for objects that are either on a site, not on a site, or have a tape copy on a site. This is useful for determining, for example, which objects need to be copied to other sites.
- **Event Search** — You can search requests for a particular type of request event. The drop-down contains some commonly searched events.
- **Other Search Parameters** — You can also search on other parameters such as file name, object size, comments, object name and category, tape barcode, and media.

  You can select the same parameter (*for example media*) multiple times and provide different values. By default, these multiple values are joined with all other values in the query using a logical AND. For example, if you search on Sitename=Site A and Sitename=Site B, your search will return objects that are present on both *Site A* and *Site B*.

  However, if you select the **OR Like Query Params** check box, the multiple values are joined together using a logical OR. For example, you can search for objects that have a media of POST1 OR PLAYOUT (*or both*).
- **Checksum Failures** — You can select to display Checksum Failures. This returns all assets that have encountered checksum failures as part of DIVA Core operations, such as **Restore** and **Copy**. The particular site that experienced the failure is marked with an alert icon in the site table. There are no query parameters for this search (*use the default value of true*).
- **Missed Deletes** — You can select to display Missed Deletes. This returns all assets where a delete was attempted, but could not be processed. The sites that experienced the missed delete are marked with a gray X in the site table. You provide a time duration in days, which is subtracted from the current date and used to filter the query to return only the most recent missed deletes.

- **Has AutoCopy Task** — You can select all objects that have an AutoCopy task associated by setting the value to "True". Setting the value to "False" will select objects that have no tasks.
- **Ordering** — You can sort the objects based on Object Name (*the default*), Category, Archive Date, and Object Size. You may see significantly slower query performance when using some optional sort methods.

Cutting and Pasting Asset Details

Many of the values in the Asset Details panel can be copied to the clipboard by double-clicking on the values. In the Asset Search panel, right-clicking on a text field reveals a menu. This menu contains a paste option, which allows items from the clipboard to be pasted into the text field. Alternatively, you can use Ctrl-V to paste the information.

Viewing Requests

Click the **Requests** tab to display a table of DIVA Connect requests in the left panel. This table is ordered by the date that each request was submitted. Each request in the table shows where content is moving from, and where it is moving to (*if applicable*). A label with an (*SD*) appended means that the content is moving from or to a Server. Otherwise, the content is moving from or to a DIVA Connect site.

Viewing Request Progress (*Live Updates*)

Requests are enabled for live updates when DIVAConnectUI starts. If the request that is queried is currently executing, the progress of the request will be automatically updated on the screen. Filters can be applied to reduce the number of requests that are dynamically updated on the screen. Live progress updates can be stopped by providing an end date and time in the Filter panel. See Filtering Requests.

Viewing Request Details

You can view the details of the request by clicking a request in the table. The details include information such as the priority of the request, the quality of service, source and target sitenames, and media (*if applicable*). In addition, a list of request-level events are displayed. These events show the steps that DIVA Connect takes to satisfy the request, and the errors encountered along the way.

Canceling Requests

Right-clicking a request in the request table will show the ***Cancel*** option. If a request is canceled, another request (*a Cancel request*) is generated that will show the status of that Cancel attempt.

Filtering Requests

DIVAConnectUI contains powerful request filter capabilities. Click the **Filter** button on the left to open the Request Filter panel.

The Request Filter panel contains parameters used to filter the list of requests. Two of the parameters are user-selectable. Each value entered poses an additional restriction on the assets returned. Wildcards are permitted in all search fields except the date and time fields and ***Request ID*** field.

> **Note:** Unlike the Asset Search panel, you cannot search on the same attribute more than once.

The following are some search terms that can be applied to the list of requests:

- **Request ID** — DIVA Connect requests are uniquely identified by a Request ID. DIVA Connect allows filtering on Request ID or the name of the object being processed.
- **Request Submission Date** — A start and end date and time range can be specified that returns only requests that have been submitted with the date range specified. Click the calendar icon to select a date. The current local (*computer*) time is automatically populated in the text field. You can modify the time value in the text field to the desired time. If an end date and time is supplied, no status or progress updates will be provided on screen. Delete the end date and time to enable live updates again.
- **Request Type** — Filters can be created on the type of request. Each type can be individually enabled or disabled. In this way, filters can ensure (*for example*) that only Archives and Deletes appear in the list.
- **Request Status** — Filters can be created on the Request Status. In this way, filters can ensure (*for example*) that only requests that are either *Pending* or *Running* are displayed.
- **Other Search Parameters** — You can also filter on other parameters such as ***Media***, ***Source/Destination, Object Name, Category,*** and ***Comments***. Finally, you can search for requests that were generated by an AutoCopy task.

Cutting and Pasting Request Details

Many of the values in the Request Details panel can be copied to the clipboard by double-clicking on the values. In the Request Filter panel, right-clicking on a text field reveals a menu. This menu contains a Paste option, which allows items from the clipboard to be pasted into the text field. Alternatively you can use Ctrl-V to paste the information.

Copying an Object

DIVA Connect can copy objects from one site to another, or it can copy the object to different archive media on the same site (*by creating a new object instance*).

1. Within the Asset Table, select the objects to copy.
2. Above the Asset Table, select **Copy** from the drop-down menu.

   A dialog box will appear showing the selected objects.
3. Select a ***Target Site***, ***Media/Storage Plan***, and ***Priority***.
   - If the object already exists on the site you are copying to, DIVA Connect will create another instance with the archive media that you select. In this case, specifying a ***Media/Storage Plan*** will cause the copy request for that object to fail, as the object on that site already has a Storage Plan.
   - If you select ***Selected By DIVA Connect*** for the ***Media/Storage Plan*** option, DIVA Connect will choose an appropriate media for the copy. If the object already exists on the site you are copying to, DIVA Connect will not create another instance on that site.
   - You can choose to run the request immediately, or place it on the AutoCopy queue for scheduling.

4. Select whether the copy should be run right away, or sent to the AutoCopy queue for batch execution.

   Click **Submit**.

5. Click either **Monitor Requests** or **Close**.

   If you choose **Monitor Requests**, you will be taken to the Requests View where you can monitor the DIVA Connect Copy requests for each object that you selected.

Deleting an Object

You can delete an object from a given site, from all sites, or from a specific media on a site.

1. Within the Asset Table, select the objects to delete.
2. Above the Asset Table, select **Delete**.
3. Select a ***Target Site***, ***Media***, and ***Priority***.

   You can select a specific site, a specific media, or all sites.

4. Click **Submit**.
5. Click either **Monitor Requests, View Assets in AutoCopyQueue,** or **Close**.

   If you choose **Monitor Requests**, you will be taken to the Requests View where you can monitor the DIVA Connect Delete requests. If you choose View Assets in AutoCopy Queue, you will be taken back to the Assets View, but will filter on assets having AutoCopy tasks.

Restoring an Object

Objects can be restored to a particular Server using the following procedure:

1. Within the Asset Table, select the objects to restore.
2. Above the Asset Table, select **Restore**. A dialog box appears showing the selected objects.
3. Enter a **Server** and select a **Priority**.
4. Optionally, enter restore options in the ***Options*** field, and (*or*) a directory path destination relative to the **Server**.
5. Check the box if you would like DIVA Connect to create a top-level directory for each object.
6. Choose whether you want to run the restore right away, or send it to the AutoCopy queue for batch execution.
7. Click either **Monitor Requests** or **Close**.

   If you choose **Monitor Requests**, you will be taken to the Requests View where you can monitor the DIVA Connect Restore requests for each object selected.

Canceling Tasks

You can delete/cancel tasks that have been scheduled on assets. Select one or more assets, and select the Cancel Tasks action. You have the option of canceling all tasks associated with the asset, or limit it to a particular target site or media (*or both*).

Modifying the Configuration File

The standard DIVAConnectUI startup script is configured to load a properties file upon startup. You can find this configuration file within the DIVA Connect home directory (*Program/conf/divaconnect/divaconnectui.properties*).

> **Note:** DIVAConnectUI needs write permissions on the properties file, as it records each change to the URL or username directly in the properties file.

You can configure a default URL in this file. This URL should refer to the ClientAdapter's network address and WebServicePort (*configured in ClientAdapter.xml*). Local web servers and routers may change the address and (*or*) port needed in the URL.

In this file, you can configure the following parameters:

- **logLevel** — DIVAConnectUI writes a log file, and you can change the default logging level by changing the *logLevel* parameter.
- **divaconnectUrl** — This represents the url that connects to DIVA Connect (*the ClientAdapter service*). Every time the url is changed in the application, it is updated here in the configuration file.
- **numObjectsToQuery** — The maximum number of objects *per page* to retrieve in the Assets View.
- **numRequestsToQuery** — The maximum number of requests to retrieve in the Requests View.
- **username** — The user name *(for example, diva*) to pass to DIVA Connect, which can be used in Access Rules.
- **userlogin** — This value is true if the UI should prompt for a username/password before attempting to connect (*default: false*).

6

DIVA Connect Administration

This chapter discusses administration of DIVA Connect.

Using the DIVAConnectAdmin Utility

DIVAConnectAdmin is a simple command-line tool that has options for administering DIVA Connect. You can only run this utility on the server where the ClientAdapter and DbSync services are running.

To start the utility, open a command window, and then navigate from the DIVA Connect home directory to the Program/divaconnect/bin subdirectory. Type one of the following commands to start the utility:

Windows:

```
DIVAConnectAdmin.bat {ClientAdapterWebUrl} {DbSyncWebUrl}
```

Linux:

```
DIVAConnectAdmin.sh {ClientAdapterWebUrl} {DbSyncWebUrl}
```

> **Note:** If you do not supply the URL arguments, the utility will assume that the ClientAdapter and DbSync services are running locally, and will prompt for the ClientAdapter and DbSync service ports.

Table 6–1 shows the DIVAConnectAdmin options, selectable through a menu system.

Table 6–1 DIVAConnectAdmin Options

Top-Level Menu Item	Tasks	Description
DIVA Connect Database...	Create Initial DIVA Connect DB	Runs sqlplus scripts to create a DIVA Connect database schema.
DIVA Connect Database...	Add Sites	Adds one or more site definitions to the DIVA Connect database.
DIVA Connect Database...	UpgradeDB Utility	Utility to upgrade the DIVA Connect database to the latest revision.
DIVA Connect Database...	Analyze Schema	Runs a script to analyze the DIVA Connect database. Useful after a large initial DBSync run.
DIVA Connect Database...	Set Tuning Parameters	Sets database tuning parameters for specific datasets.

Table 6–1 (Cont.) DIVAConnectAdmin Options

Top-Level Menu Item	Tasks	Description
DBSync...	List Synced Sites	Generates the list of the sites in which asset information is being synchronized. It will also display the synchronization status of each site. If a site is in an initial sync state, the percentage complete will also be displayed.
DBSync...	Resync a Site	Resynchronizes all object records from a particular site to the DIVA Connect database. **Caution: This option will delete all asset records in the DIVA Connect database (*for the selected site*) before resyncing them. In addition, AutoCopy rules will be run again on each resynchronized object.**
DBSync...	Resync an Object	Updates object records for a specific object at a specific site.
Sites...	Show Site Parameters	Displays DIVA site parameters, including the number of queued requests to the site, number of requests (*sent by DIVA Connect*) running on each site, date of last communication with the site, and the throttling parameters.
Sites...	Set Site Parameters	Allows the user to set the DIVA throttling parameters: • The *SubmittedThrottleThreshold* controls how many requests DIVA should send to a site before waiting (*for one of them to complete*). • The *TotalThrottleThreshold* is similar, but is based on how many total requests are running on the DIVA site. DIVA Connect will wait until the total DIVA requests drops below this number (*before sending the next request*). If Access Rules are defined, this option requires access to the ChangeConfig operation.
Sites...	Add Sites	Adds one or more site definitions to the DIVA Connect database.
Services...	List DIVA Connect Services/Status	Lists the DIVA Connect services that were installed from this installation directory.
Services...	Start/Restart a DIVA Connect Service	Starts (*or restarts*) a DIVA Connect service.
Services...	Stop a Service	Stops a DIVA Connect service.
Services...	Install a new DIVA Connect Service	Installs a DIVA Connect service.
Services...	Uninstall a Service	Uninstalls a DIVA Connect service.
AutoCopy...	Pause AutoCopy Scheduling	Pauses AutoCopy scheduling. This has no effect on requests that have been created from the tasks.
AutoCopy...	Unpause AutoCopy Scheduling	Unpauses AutoCopy scheduling.
AutoCopy...	Clear Entire Queue	Clears the entire AutoCopy queue.
AutoCopy...	Evaluate Rules on Existing Assets	Evaluates the AutoCopy rules on all existing assets. Take care with this option, as thousands of tasks could be generated.
Reload Workflow Profiles/Rules	Reload Workflow Profiles	Causes all Workflow Profiles to reload. It will also reload any Access Rules or AutoCopy Rules that are defined. If Access Rules are defined, this option requires access to the ChangeConfig operation.

Table 6–1 (Cont.) DIVAConnectAdmin Options

Top-Level Menu Item	Tasks	Description
Logs...	Get Log Level for ClientAdapter	Lists the log level of the ClientAdapter services.
Logs...	Set Log Level for ClientAdapter	Dynamically changes the log level (*valid values are* ***ERROR, WARN, INFO, DEBUG, TRACE***).
Logs...	Get Log Level for DbSync	Lists the log level of the DbSync service.
Logs...	Set Log Level for DbSync	Dynamically changes the log level (*valid values are* ***ERROR, WARN, INFO, DEBUG, TRACE***).
Users...	Add a User	Adds a DIVA Connect user and password to the DIVA Connect database. Insure that the ClientAdapter parameter UserAuthEnabled is set to true, and set the user's DIVA Connect UI property file parameter userlogin to true.
Users...	Delete a User	Delete a user from the DIVA Connect database.
Users...	List Users	List the users that have been created.
Users...	Change Password	Changes the user's password. The password will expire in 180 days.
Misc...	List API Socket Connections	Lists all Client DIVA Connect API connections, including those in Direct Mode.
Misc...	Kill an API Socket Connection	Provides an option to end a particular API connection, identified by ID.
Misc...	Generate Unique ID	Generates a unique id that can be used as a username, for example.
Misc...	List Cleanup Statistics	Lists statistics for the assets that need to be purged after transfers occur.

DIVA Connect creates the following types of log files:

- **Trace Logs** — Located within the DIVA Connect home directory in the Program/log/divaconnect subfolder. Each of the DIVA Connect services (*ManagerAdapter, DbSync, and ClientAdapter*) create their own subfolder underneath the Program/log/divaconnect folder and generate their log files in that subdirectory.
- **Wrapper Service Logs** — These logs output messages related to startup and shutdown of the windows service. This is the place to look if the service is not starting correctly. These logs are within the DIVA Connect home directory in the Program/log/divaconnect subfolder. They will be named similar to the Wrapper.conf file, but with a .log file name extension (*for example, ManagerAdapterWrapper.log*).
- **GUI Logs** — Errors that occur in DIVAConnectUI will appear on the client computer, not on the server side. The logs appear in the Program/log/divaconnect/gui subfolder.

DIVA Connect will package logs older than one hour into zip files, and retain two days worth of logs for review.

Request Processing and Queuing

The following sections discuss queuing and processing of DIVA Connect requests.

DIVA Connect Request Queuing

DIVA Connect can accept more requests than downstream DIVA Core systems can process with request queues. DIVA Connect enforces a maximum number of requests that can be queued and (*or*) running internally (*the default is 5000*), using the GlobalDivaconnectRequestLimit parameter. DIVA Connect will start rejecting requests if this limit is reached. The DIVA Connect ClientAdapter service maintains a queue for each site. You can configure DIVA Connect to feed requests to sites slowly, or to send requests to sites all at once (*where they will likely be queued by DIVA Core*).

If a request is queued to a DIVA Core system, and the DIVA system is restarted, all running requests will be terminated. DIVA Connect behaves differently - DIVA Connect stores each request in a database. If the ClientAdapter service is restarted, DIVA Connect reads the pending and running requests and restarts them. However, not all requests will be restarted this way. For example, delete requests will never be reissued after the service restarts. If recovering requests on startup is not desired, the ClientAdapter AbortAllOnStartup parameter can be set to terminate all requests in the queue at startup.

DIVA Connect requests are distinct from tasks in the AutoCopy queue. For more on Copy requests and AutoCopy tasks, see "AutoCopy" on page 4-26

DIVA Connect Request Throttling

By default, DIVA Connect will send one hundred requests to a site before stopping and waiting for one of the requests to complete. The parameter that controls the limit is SubmittedThrottleThreshold, which is configurable in the ClientAdapter on a per site basis. It can also be changed at run-time through the DIVAConnectAdmin utility. In addition, DIVA Connect can throttle requests based on the total number of requests running on a DIVA Core site (*using the TotalThrottleThreshold parameter*). You can set either parameter to **-1** to indicate that no throttling should be performed. You can also set the value to **0**, indicating that no requests in the queue should be sent to DIVA Core. This can be useful in certain situations.

If a DIVA Connect request is queued to a particular site for more than a few seconds, DIVAConnectUI will display the position of the request in the queue. If another request is queued with a higher priority than the first, it will likely have a higher position in the queue. If the request fails and is retried, the request will be queued again (*although it will get a priority boost from being in the system for a long time*).

Delete requests associated with queuing and timeouts are treated differently than other requests. The DeleteWaitTimeoutMins parameter controls how long DIVA Connect waits for a delete to complete before timing out the operation. The default is two minutes, but can be increased to be more in line with other commands.

DIVA Connect Retries

Some DIVA Connect operations can be retried automatically within DIVA Connect. There are two type of retries in DIVA Connect. The first are AutoCopy retries on Copy or Restore tasks. AutoCopy will automatically spawn new requests when one fails, until a request is successful. The other type of retries are applied at the Workflow Profile level -- without respawning a new request. With Workflow Profile retries, Copy, Restore, Partial File Restore, and Delete requests can be retried. Restore & Partial File Restore requests can be retried a configurable number of times, where Copy and Delete requests are retried over a configurable time interval. These retries are modified at run-time without restarting the ClientAdapter (*see* Configuring Workflow Profiles *for more information*).

If a site is down, and requests are queued to that site, DIVA Connect will wait for SiteDownRequeueWaitMins minutes for the site to become operational. If that configured time period elapses, and retries are configured, DIVA Connect will attempt to retry the request, possibly on another site. If retries are *not* configured the request will terminate.

7

DIVA API

This chapter highlights DIVA Connect's DIVA API support, and provides information on how to supply DIVA Connect specific information using the DIVA Core API. This chapter is intended to be used with either the DIVA Core C++ API, DIVA Core Java API, DIVA Core Web Service API documentation, or the new REST API (*documented using Swagger*).

DIVA API credentials can be used to authenticate users configured in the DIVA Connect database. Users configured with a Workflow Profile will use that profile for the session. This feature is only available on secure C++ and Java API connections and with the REST API. Use <UserAuthEnabled> in the ClientAdapter <Ports> section. See Configuring Client API Ports for more information on how to set up authorized API connections, and *Using the DIVAConnectAdmin Utility*, Table 6–1, *Task: Add a User*, for information on adding users to the database.

- DIVA Connect API Version Compatibility
- Operations on Archived Content
- Other Supported Commands
- Unsupported Requests
- Return Codes

DIVA Connect API Version Compatibility

DIVA Connect supports a subset of the full DIVA Core API command set. Some DIVA Core API commands (*such as EjectTape*) will be rejected in DIVA Connect MultiDiva Mode. DIVA Connect does support client connections from DIVA Core API clients releases 7.3 and earlier. Any new features added to the API after 7.3 will not supported by DIVA Connect.

Invoking DIVA API calls against a DIVA Connect server is largely the same as invoking calls against DIVA Core. However, there are some differences - DIVA Connect sometimes accepts the well-known DIVA API parameters in a slightly different way. In addition, the content of fields returned by DIVA Connect may be slightly different from DIVA Core, or have a different format. This section highlights those differences.

DIVA Connect requires that connected DIVA Core sites be installed with DIVA Core 7.3.1 or later. DIVA Core sites may then be upgraded independently without requiring an upgrade to DIVA Connect.

Operations on Archived Content

This section highlights requests that involve the transfer (*or deletion*) of archived content, including Archive, Restore, Delete, and Copy. These commands can be invoked through the DIVA Core API. Some can also be invoked from the DIVAConnectUI. *Refer to the DIVA Core C++*

API Programmer's Guide in the DIVA Core Documentation library for more information on what each command within the DIVA Core API does.

Note: In DIVA Connect MultiDiva Mode, DIVA Connect requests often require a few additional pieces of information that are not needed by other requests issued directly to DIVA.

DIVA Connect requests often require additional information because DIVA Connect can access multiple sites. For instance, you can use the Copy command in DIVA Connect to copy content from one DIVA system to another. DIVA Connect needs to know, at a minimum, what the target site is. However, the DIVA API CopyToGroup command does not contain a *target site* parameter. The sections below detail how to specify this additional information. For more information on configuring DIVA Connect MultiDiva Mode, see Configuring Client API Ports.

Table 7–1 identifies the DIVA Core API Content Requests that are supported by DIVA Connect. Clients that issue these requests are provided a Request ID in return, which can periodically query the status of the request.

Table 7–1 Supported DIVA Core Content Requests

Request	Behavior in DIVA Connect
Archive	Enables digital assets from a Server (*such as FTP or CIFS file system*) to be transferred to, and stored in, the local DIVA Core site, or optionally, another chosen site. For more information, see Archive Requests.
Restore Restore Instance	Enables a DIVA Connect object to be restored (*transferred*) to a Server (*such as FTP or CIFS*). You can specify the site and (*or*) instance to restore, or let DIVA Connect choose the best site. DIVA Connect decides whether it should (*1*) restore an object from the local DIVA Core, (*2*) directly restore an object using a remote DIVA Core, or (*3*) pull an object from a remote DIVA Core and then transfer it to the chosen Server. A Restore Instance allows a specific instance at a specific site to be restored. If a failure occurs, you can configure DIVA Connect to retry on other sites. For more information, see Restore Requests.
Partial Restore Partial Restore Instance	Partially restore an object in a manner similar to a full restore. If issuing a partial file restore on a remote system, that DIVA site must be configured for partial restore in a manner similar to the local site. For more information, see Partial File Restore Requests.
Copy (*CopyToGroup*)	Copy content from one DIVA site to another (*an Intersite Copy*), or create another instance of an object on new media at a DIVA site (*equivalent to issuing a CopyToGroup on a single DIVA system*). A specific instance can be copied to a target site. You can configure DIVA Connect to retry for a given time period. For more information, see Copy Requests.

Table 7–1 (Cont.) Supported DIVA Core Content Requests

Request	Behavior in DIVA Connect
Delete DeleteInstance	Delete content from all sites, a specific site, or delete a specific instance on a specific site. If objects are locked on sites to be deleted, you can configure DIVA Connect to retry for a given time period. For more information, see Delete Requests.

Archive Requests

An Archive request transfers content from a Server (*configured in DIVA Core*) to a DIVA Core media (*disk, tape, or cloud*). The API caller arranges for the transfer to be made, but does not actually transmit the content. The DIVA API arranges for a transfer to occur from the Server to DIVA Core. This differs from a web-based cloud API where the content is transferred directly from the requester through HTTPS. By default, DIVA Connect archives to the local site.

Archive requests issued to DIVA Connect are similar to those directly to DIVA Core, but with a few additions. Normally, DIVA Connect will archive to the local site. However, you can archive directly to another site in one of two ways:

- By providing the **-site** {sitename} option in the options field. An example would be **-site** *diva1*.
- By prefixing a Destination Server sitename to the media parameter in the archive request. For example, *sitename1_TapeGroup1* indicates a Destination Server site called *sitename1*, and a media of *TapeGroup1*.

DIVA Connect does not support continuous retries for Archive commands, but does support a BackupArchiveSite option, which provides an alternate archive site in case the primary site is down.

Restore Requests

A Restore request allows the client to restore content that exists in the archive system. The content arrives at the particular Server chosen in the request. The DIVA API arranges a transfer from a DIVA Core site directly to a Server (*such as FTP or CIFS disk*). This differs from a web-based cloud API where the content is transferred directly to the requester through HTTPS.

When restoring content using DIVA Connect, the caller need not know which particular DIVA system has the content. And if failures occur retrieving content from one DIVA site, another DIVA site can automatically be consulted to retrieve the content.

DIVA Connect supports restoring to any Server on any DIVA Connect site. DIVA Connect will retrieve content from other sites as needed to satisfy the request, ultimately transferring the content to the target Server.

Restore requests issued to DIVA Connect are similar to those issued directly to DIVA Core, but with a few additions. Normally, DIVA Connect chooses the best site to restore from. However, DIVA Connect enables a client to specify the specific site to use for the restore. There are several ways this can be done:

- **-site**: Normally, DIVA Connect will choose the site from which to restore. However, you can attempt a restore from a particular site by providing the **-site** *[sitename]* option in the options field of the request. If in fact the content is not located at the chosen site, the operation will fail.
- **Instance Id**: If you need complete control over the Source Server, you can provide an instance number in the restore request. This will allow you to choose the Source Server site, and the DIVA instance to restore from (*see the following section*). This instance id can

be obtained by performing a getObjectInfo() API call, or by viewing the object in DIVA Connect UI.

In both of these cases, retries are disabled.

Restore Workflow

To satisfy a restore request, DIVA Connect uses the restore methods in Table 7–2. DIVA Connect will dynamically select which restore workflow to use based on parameters such as the target Server and the source object. To decide which site to use for restores, DIVA Connect asks a series of questions, including:

- Is the object available on the local DIVA Core System?
- Does the object have a disk instance?
- Is the Server accessible from the remote DIVA Core System?
- Is the Server accessible from the local site?
- Is DIVA Core running on the Source Server or Destination Server site?
- Is one site preferred over another in the configuration file?

Table 7–2 DIVA Connect Restore Methods

Method	Description
Local	Used when an object exists on the local site. The local site is the DIVA Core System sitename to which you are sending messages. A local DIVA Core System is also considered part of the local site.
Direct Remote	DIVA Connect can arrange to have a remote DIVA system perform a restore operation. It will do this only if the target Server is also configured in the remote DIVA system. The Server names must match, and they must both refer to the same server or disk (*and path on that disk, if applicable*). When available, DIVA Connect prefers this method over performing a Restore using Intersite Copy.
Using Intersite Copy	If content is not local, and if a remote DIVA system cannot directly restore to the target Server, DIVA Connect can arrange for the content to be delivered in two hops. First, the remote DIVA restores to a Server shared between the source and target sites. Then, the local DIVA will archive the object, and finally, restore to the target Server. This way future requests for the content will be retrieved much quicker. If it is desired to always perform remote restores by creating a nearline copy, set AllowDirectRemoteRestores to false in the Workflow Profile.
Using Intersite Transfer	In certain cases, when DIVA Connect is not able to perform a direct remote restore, DIVA Connect will deliver the content in two hops (*as Restore using Intersite Copy*) but will not actually archive the content locally. One such case involves Partial File Restore. First, DIVA Connect will instruct the Source Server DIVA to transfer the content to the Server accessible by both the source and target DIVA sites. Then, the local DIVA site will transfer the content to the target Server without archiving it.

Table 7–3 DIVA Connect Restore Options

Attribute	Description
Options	• -autocopy: Providing this option adds the request as a task on the AutoCopy queue. This means that the request is not immediately executed. Instead, AutoCopy schedules the task and ensures that it completes. • -autodelaymins: If -autocopy appears, this parameter delays the execution of the task by at least this amount (*in minutes*). • -site: This option will indicate a site to be used as the source for the restore. Normally, DIVA Connect will select the best site for the restore.

Source and Destination Servers

DIVA Connect will allow you to restore to a Server that is available on any site. DIVA Connect assumes that if a Server exists with the same name on two sites, both configurations point to the same physical Server\Device\Path. Users must be careful with the names that are assigned to Servers across the sites.

If a remote DIVA system cannot directly restore to the target Server, DIVA Connect can arrange for the content to be delivered in two hops (*see Restore Using Intersite Transfer*). DIVA Connect will arrange this only if the Server is the preferred site for the particular Server (*see* Preferred Server Mappings).

Retries

You can configure the Restore command to retry several times if the first restore fails. If the content to be restored exists on multiple sites, DIVA Connect will automatically retry the restore with those sites. You can configure the maximum number of retries. In some cases, DIVA Connect will decide to retry with the same site before moving on to other sites. In this case, DIVA Connect will examine the RestoreRetryIntervalMins value to determine how long to wait before retrying with the same site.

Limitations

- DIVA Connect supports a subset of the DIVA Core API. Refer to this chapter for a list of the supported messages.
- DIVA Connect has limited support for Multi Restore. DIVA Connect does not allow Multi Restores for remote Servers, and does not allow viewing or monitoring of multiple Servers through the UI.
- DIVA Connect does not support peer-to-peer site relationships, including load balancing.

Partial File Restore Requests

In addition to full restores of content, DIVA Connect also supports Partial File Restores. DIVA Connect determines the site where the content is located, and arranges to have the portion restored.

If an object exists on a remote DIVA system, and the target Server for the restore is not accessible by a remote DIVA system, DIVA Connect transfers the content in two steps - first using the remote DIVA to get the content to the local DIVA (*without transferring the entire object*), and then using the local DIVA to restore the content to the target Server.

As with restore, you can specify the instance number or **-site** parameter to perform a restore from a specific site, or specify retries when the initial site fails. And also as with restore, restoring to multiple Destination Servers within the same restore request is not supported.

Copy Requests

A Copy request creates a new instance of archived content from an existing instance. DIVA Connect allows content to be copied from one DIVA site to another. The DIVA Core API CopyToGroup command (*1*) copies an object from one DIVA site to another, or (*2*) simply creates a new instance on a single site. For copies, DIVA Connect needs to derive some parameters that are not available in the DIVA API. Table 7–4 describes these parameters.

Table 7–4 DIVA Connect Copy Derived Parameters

Derived Attribute	**Description**
Target Sitename	Indicates which site the object should be copied to. Target sitename does not exist as an architected field in the DIVA API. You can convey this to DIVA Connect in one of two ways: • **-site Option**: By providing the **-site** [*sitename*] option in the options field. An example would be **-site** *sitename1*. • **Sitename Media Prefix**: By prefixing a Destination Server sitename to the media parameter in a copy request (*with an underscore (_) separating the two fields*). For example, *sitename1_TapeGroup1* indicates a copy Destination Server site called *sitename1*, and a media of *TapeGroup1*). If you specify no sitename, the local site will be assumed. The **-site** keyword will work only with an API release 7.3 or greater.
Additional Options	• -autocopy: Providing this option adds the request as a task on the AutoCopy queue. This means that the request is not immediately executed. Instead, AutoCopy schedules the task and insures that it completes. • -autodelaymins: If -autocopy appears, this parameter delays the execution of the task by at least this amount (*in minutes*). • -replaceTarget: If the object already exists on the target site, delete the content and re-archive the object. • -fromSite: The site to copy from • -site: The site to copy to
Media	**Media** indicates the type of media to use to store the copied object. DIVA Connect also enables providing a DIVA Storage Plan as a media name. **Storage Plan** will work only if the copy is an intersite copy. You can prefix the target sitename to the media to indicate the target sitename as well. If you are not sure which media to provide, you can provide a media of **any** to let the system choose which media to store to on the target site. For example, a media of *diva1_any* copies to site *diva1*, but DIVA Connect chooses the media. The default that DIVA Connect provides may not be appropriate for some use cases. If the object is already on the target site, and **any** is specified, the system will simply return success.

Table 7–4 (Cont.) DIVA Connect Copy Derived Parameters

Derived Attribute	Description
Source Sitename	Normally, DIVA Connect will choose the site to copy from. However, if you need complete control over the Source Server, you can provide an instance number in the copy request. This implicitly enables choosing the Source Server site, and which DIVA instance to copy from (*see the following section*). This id can be obtained by performing a getObjectInfo() API call on the object to be copied. Additionally, -fromSite can be passed to indicate the specific site to copy from.

Site-to-Site Transfer Methods

In a DIVA Connect Copy request, if the Source Server site is the same as the target site, DIVA Connect can simply issue a CopyToGroup to the target DIVA site. For intersite copies, DIVA Connect enables configuring the method used to perform these copies. For each source and target sitename pair (*for example, site1 to site2*), Table 7–5 lists the available transfer methods.

In addition to the copy method, each source and target sitename pairing contains the actual Servers used for the common storage area. The default destination media (*for* ***RestoreAndArchive***), options parameters (*passed to DIVA Core*), and other parameters are also configurable.

Table 7–5 Site-to-Site Copy Methods

Type	Description
RestoreAndArchive	With this option, DIVA Connect restores content from the source site, to a Server that is common to the source and target sites. Then, DIVA Connect instructs the target DIVA Core System to archive the content now in the common storage area. This is an alternative to using Drop Folders.
RestoreAndMonitor	Using this method, DIVA Connect will perform the copy by first restoring the content to a specific Destination Server. DIVA Connect will then turn to monitoring the target DIVA Core system to determine when the content is successfully archived on the target site. The request will successfully complete only when the content is successfully archived on the target site. This of course relies on another process or program that will take the content and archive it into the target DIVA system. This option is useful in combination with DIVA Core DFM (*Drop Folder Monitor*) software. Each DFM folder is configured to archive using a preselected media - meaning that when DFM is employed for copies, the media parameter is effectively ignored.
Restore	Using this method, DIVA Connect will perform the copy by restoring to a specific Destination Server, and then returning success. This method does not Confirm that the content has been successfully archived into the target site, and will likely cause failures if Restore Using Intersite Copy workflows are attempted.

Table 7–5 (Cont.) Site-to-Site Copy Methods

Type	Description
ActorSiteTransfer	This method uses the Datahub to Datahub feature of DIVA. The workflow is similar to RestoreAndArchive, except the Restore from the Source Server and the Archive to the Destination Server are executed simultaneously. The transfer takes place between Datahubs on the two sites. No temporary instance is created. This option requires that a HolePunchServer service is running and usable by both sites. The Server must be specified as diva_bridge.

The DIVA Connect Copy command will return success if an instance of the object already exists in the target site on the requested media. DIVA Core will terminate the request in this case.

Local Copies

If an object is requested to be copied to a site where the object already exists, but does not have the requested media, DIVA Connect will create another instance of the object on that site using the media specified in the request. The exception to this is if **any** is supplied as the media. In this case, DIVA Connect will not create another instance.

Selected By DIVA Connect (*media of any*)

In DIVAConnectUI, there is an option to have DIVA Connect assign the target media on a Copy operation (*use a media of Selected By DIVA Connect*). You can accomplish the same thing in a DIVA API request by specifying a media of **any** in the Copy request. DIVA Connect will use its configuration to determine which media to use for the copy (*see* Site-to-Site Mappings *for more information*).

When **any** is passed, and the object already exists on the target site, DIVA Connect will not create another instance of the object. No further action will be taken, and the request will succeed.

Retries

DIVA Connect also supports periodic retries on copies. If enabled in the Workflow Profile, DIVA Connect will retry copy operations that failed. In the Workflow Profile, you can configure how long DIVA Connect will retry the request, and the wait period between retries. See Configuring Workflow Profiles for more information.

Delete Requests

A DIVA Core Delete request allows the caller to remove an archived object. A DIVA Connect Delete request, by default, will delete the object from all DIVA Core sites. A DIVA Connect DeleteInstance request can delete one or all instances from a single DIVA site. So in reality, DIVA Connect can perform three kinds of deletes. Table 7–6 describes the delete types and parameters they require.

Table 7–6 Delete Types

Type	Derived Attributes	Description
Global Delete	NA	Deletes all object instances on all sites. In the API, not specifying a media or instance id in the request will cause the specified object to be deleted from all sites.
Instance Delete	Target Sitename	Deletes a single object instance on a particular site. In the API, specify an instance id or a media to delete a specific object instance from a specific site. If you specify an instance id, you are targeting a specific instance on a specific site. Passing media or options parameters are not necessary. If you specify a media, DIVA Connect needs to know the site you would like to delete from. You can specify the site in one of three ways: • **Media Sitename Prefix**: You can prefix the sitename to the media. For example, *siteA_tapeGroup1* tells DIVA Connect to delete the instance on *siteA* having a media of *tapeGroup1*. If you do not provide a sitename, the local site will be assumed. • **-site Option**: Provide **-site** [*sitename*] in the options field, and supply a media. This is supported with DIVA Core API 7.3 or later. • **Media (*with no site*)**: You pass the media with no site. In this case, the local site will be assumed.
Site Delete	Target Sitename	Deletes all instances of an object residing on a particular site. This can be accomplished in one of three ways: • **Use DIVAConnectUI**: You can delete all instances on a particular site by issuing a Delete command from the DIVAConnectUI. • **Sitename/Media of any**: Issue DeleteInstance through the API by providing the sitename and a media of **any** in the media field (*for example, siteA_any*). • **-site Option**: Specify a sitename by providing **-site** [*sitename*] in the options field. This is supported with DIVA Core API 7.3 or later.

Site Delete (*media of any*)

As mentioned in the previous table, if a sitename is supplied and a media of **any**, this will cause all instances of the object to be deleted from the selected site. This can also be accomplished by passing **-site** [*sitename*] in the options field.

Last Instance

DIVA Connect will not allow an Instance Delete to succeed when attempting to delete the last instance of an archived object (*the last instance that exists in the DIVA Connect database, that is*). In this case a Global Delete or Site Delete request will need to be issued. Note however that a Site Delete will allow the last instance or instances to be deleted! You can use Access Rules to protect against Instance Deletes or Site Deletes that are effectively Global Deletes (*see* Access Rules *for more information*).

Also, if a user issues a delete directly to DIVA Core, there may be a time window where DIVA Connect cannot ensure that the last instance is preserved.

Retries and Cancels

If a Delete is received by DIVA Connect, and DIVA Connect is currently performing an Intersite Copy of the object, the DIVA Connect request that spawned the copy will be canceled. The copy could be the result of a Copy command, or, it could be the result of a Restore command that performs a copy to satisfy the restore. Other types of DIVA Connect requests will not be canceled.

If a DIVA Core request is being executed on behalf of a DIVA Connect request, DIVA Core locks the object and prevents it from being deleted. So, if an object is locked before DIVA Connect can send a delete message to DIVA Core, the delete request will fail.

DIVA Connect supports periodic retries on deletes when deletes fail. If enabled in the Workflow Profile, DIVA Connect will keep trying to delete on sites where (*for example*) instances/objects to be deleted are locked. You can configure within DIVA Connect how long to retry.

Other Supported Commands

Table 7–7 identifies DIVA Core API commands that do not specifically process or transfer archived content. These commands obtain information about objects or requests and are not assigned a request ID.

Table 7–7 Other Supported DIVA Core Commands Not Related to Content

Request	Description	Behavior in DIVA Connect
Cancel	Cancel a DIVA Connect request.	The **-site** option does not apply for this command.
GetObjectInfo	Use the DIVA Connect Database to get information about an archived object. DIVA Connect returns all instances of the object on all DIVA Connect sites. You provide the Object Name and Object Category DIVA Connect uses the DIVA Connect database to return information about the archived object. In the GetObjectInfo() response, DIVA Connect prepends the sitename to the media parameter of each instance of an object.	The **-site** option is not supported for this command.
GetRequestInfo	Retrieve information regarding a DIVA Connect request from the DIVA Connect database. **Note**: The Additional Information parameter has limitations. The additional information is provided by DIVA Core and represents information from the last DIVA request processed. The information does not reflect the other sites in the DIVA Connect network.	When DIVA Connect receives a GetRequestInfo() call, it provides information about DIVA Connect requests, not DIVA Core requests (*although the DIVA Connect requests often involve invoking one or more DIVA Core requests*). The **-site** option does not apply for this command.

Table 7–7 (Cont.) Other Supported DIVA Core Commands Not Related to Content

Request	Description	Behavior in DIVA Connect
GetPartialRestoreRequestInfo	Returns Partial Restore offsets for a completed request. If the format for the offsets was specified as timecodes, the offsets that are actually used may differ (*somewhat*) from what was specified in the request. After the Partial File Restore request is complete, you can use this command to obtain the actual offsets of the restored files.	This is a special purpose command that is valid only as follows: - The request number to be queried must be a Partial File Restore request that has been successfully completed. - The format specified in the Partial File Restore request must be a timecode type. This command is therefore not valid when the format of the request was folder-based or DPX.
GetFilesAndFolders	Retrieve information directly from DIVA Core about the files and folders within a particular archived object.	Accepts the **-site** option to query a specific site, or no site to let DIVA Connect choose (*recommended*).
GetObjectDetailsList	Retrieve information from DIVA Core regarding objects and object events. DIVA Connect retrieves the object information directly from each DIVA Core system one site at a time, in a round-robin fashion, one batch per site. Each batch contains information from one DIVA site. If the same object exists on two sites, you will receive the object twice (*once for each site*), once in each batch. **Note**: The order of the returned entries is not guaranteed. If one DIVA site is down, GetObjectDetailsList() will return an error and processing of the command will stop.	You can retrieve information from one site by prefixing the sitename to the media field separated with an underscore (_). If you do not want to query on media, but still want to retrieve information from one site, you can supply just the sitename in the media field. This command does not accept the **-site** option.
GetObjectsList	Queries the DIVA Connect Database to get a list of Object Name and Categories (*InitObjectList, CloseObjectList*).	There is limited support for this command within DIVA Connect. Tape information queries are not supported; and there are limits on number of simultaneous queries. The **-site** option is not supported.
GetArrayList	Returns a List of Array Names from all sites including the disks that form each array, and returning the current disk capacities.	The **-site** parameter, passed in the options field, can return information for a specific site.
GetGroupsList	Returns a list of Tape Group Names from all sites. The sitename is prefixed to the Group Name.	**Note**: This command does not have an options field. As a result, the **-site** options is not supported.
GetSourceDestinationList	Returns a list of Source and Destination Server information from all sites.	In the returned list, the sitename of the Source or Destination Server is prefixed to the Source or Destination Server name separated by an underscore (_). The **-site** parameter, passed in the options field, can return information for a specific site.

Table 7–7 (Cont.) Other Supported DIVA Core Commands Not Related to Content

Request	Description	Behavior in DIVA Connect
GetArchiveSystemInfo	Retrieves the status of a single DIVA Core site (*by default, the local site is returned*). It does not return a global view of all sites.	The **-site** parameter, passed in the options field, selects which site to gather information from. For example, **-site** *diva1* will route the GetArchiveSystemInfo request to the Manager designated with the sitename *diva1* and return Archive System Information for *diva1* only.
GetStoragePlanList	Retrieves the list of storage plans that are defined in all configured DIVA Core sites.	In the returned list, the sitename of the Storage Plan is prefixed to the Storage Plan name, separated by an underscore (_). The **-site** parameter, passwd in the options field, returns information for a specific site.
GetObjectsByFilename DeleteFile	The DIVA Core AMC product uses calls (*for internal use only*). The workflows supported in this release are limited to those where all desired DIVA Core AMC objects exist on the local site. Replicas of the content can exist and be restored from other sites.	DIVA Connect invokes these commands on the local site. If the local site is down, GetObjectsByFilename is invoked on another site. Limited sequence support.
getVersion	Returns the DIVA Connect build and server versions.	
GetTapeInfo	Retrieves detailed information about a particular tape.	DIVA Connect will accept a barcode as an argument, and send this message to the local site. If a sitename is prefixed to the barcode (*separated by an underscore character*), DIVA Connect will send the request to a particular site.
TranscodeArchived	This creates a transcoded copy of a particular asset, and stores it as a new asset.	The -site parameter, passed in the options field, selects which site to route the request to (*the default is the local site*). This command cannot be used to transcode to another site.
CopyToNew	This creates a copy of a particular asset, and stores it as a new asset.	The -site parameter, passed in the options field, selects which site to route the request to (*the default is the local site*). This command cannot be used to copy to another site.

Unsupported Requests

For application compatibility reasons, these commands will always return success, even though DIVA Connect will take no action to satisfy them.

- Change Priority
- Lock Object
- Unlock Object

- Link Objects
- Require Instance
- Release Instance

Return Codes

DIVA Connect returns status codes that are similar to what DIVA Core returns. However, DIVA Connect will sometimes accept requests that DIVA Core would fail immediately, because DIVA Connect often does not have the information needed to perform checking until later during request processing.

In addition, DIVA Connect will return the ACCESS_DENIED status for many commands. This status is not returned by DIVA Core. DIVA Connect will reject requests that do not pass Access Rules checks, and rejects messages not configured in the WorkflowProfile. For compatibility purposes, API release 5.8 and earlier return INVALID_PARAMETER status instead of ACCESS_DENIED.

HTTP Return Codes

In addition to the previous return codes the REST API will provide the following HTTP return codes:

200 Success
This does not necessarily mean the operation will complete successfully. It just means all required fields were found and the operation was accepted by DIVA Connect.

400 Bad Request
An invalid parameter was found.

401 Unauthorized
DIVA Connect accepted the command, but the DC user is not authorized to perform the operation.

403 Forbidden
Most likely an incorrect or expired Bearer token.

404 Object Not Found
DIVA connect was unable to locate the object.

406 Not Acceptable
DIVA Connect did not return the expected response. This could happen if the DC user authentication is enabled, but no valid DC user is logged in.

500 Internal Server Error
This could be due to malformed request body.

8

Troubleshooting

The following are common errors that can occur in a DIVA Connect installation. For instance, although tag names in configuration files are not case sensitive, the values typically are (*for example, sitenames*), and can cause errors.

- DIVA Connect Common Errors
- DIVAConnectUI Common Errors

Table 8–1 DIVA Connect Common Errors

Problem	Possible Resolution
A DIVA Connect service does not start	• When installing a service, make sure to pass the name of the adapter's wrapper.conf file and not the name of the adapter's configuration file. • Make sure to place the configuration file in the parent directory of the templates folder, and remove the .ini extension. • If there are errors reading the configuration file, they will appear in the wrapper logs. These logs are placed in the DIVA Connect home directory, in the Program/log/divaconnect folder, and will be named similarly to the Wrapper.conf file, but with a .log file name extension. • You must run the addSites script before attempting to start the ClientAdapter or DbSync. • Are the sitenames consistent between the database and the configuration files? If not, services may not start and (*or*) not function properly.
DIVA Connect request is returning access denied	• Ensure that the correct Workflow Profile is configured for the port you are connecting on. Ensure that the request is permitted in the list of valid Messages section of the Workflow Profile. • If you are using Access Rules, ensure that **ApiConnect** appears in an Include rule. If you are using DIVAConnectUI, ensure that **WebConnect** appears.

Table 8–1 (Cont.) DIVA Connect Common Errors

Problem	Possible Resolution
Intersite copy operations fail with errors referring to Source and Destination Server names containing *MISSING_MAPPING_TO*	• Ensure that the request is being assigned the Workflow Profile that you expect by checking the configuration. • The ClientAdapter Site-To-Site Transfer configuration may be missing the from Source Server or to Destination Server.
No requests are being sent successfully to a DIVA site, even though they can be viewed in DIVA Connect	• Ensure that the *localSitename* in the ManagerAdapter configuration files correctly reflect the site that they are serving (*and are not cut and pasted from somewhere else*). Check that the *localSitename* is correct in the ClientAdapter and DbSync configurations. Check that the sitenames used are consistent with those defined in the DIVA Connect database. • Check that the network address and port of the DIVA Manager you want to send to is correct in the ClientAdapter configuration.
In Access Rules, after defining an Exclude rule, the operation is being rejected in all cases.	• Remember that for any operation to succeed, there must exist one Include rule for the operation in question. • Pay attention to any defaulted parameters in the Ruleset, such as WorkflowProfile, and change them accordingly.

Table 8–2 DIVAConnectUI Common Errors

Problem	Possible Resolution
Cannot connect to DIVA Connect	• Ensure that you have specified the correct URL of DIVA Connect. By default, this is the address of the computer that the ClientAdapter is running on, at the port defined in the *WebServicePort* parameter of the ClientAdapter configuration. • Make sure to specify the port in the url (*if applicable*), and specify *https* if secure connections are enabled.
UI is no longer updating progress of existing requests, or not showing new requests.	• Check the lower-right corner of the DIVA Connect GUI to determine if a connection error has occurred. If this is the case, click the **Requests** or **Assets** buttons to re-spawn the connect dialog box. • Perhaps you have provided an end time in your request query. This indicates to DIVAConnectUI that live updates should not be shown, and no progress should be updated for current requests. To enable live updates, just delete the end time filter. • Finally, check the Request Filters pane to ensure that there are not any Request Type or Request Status filters enabled (*likewise for Start and End time*).
Newly archived objects are not showing up in the Assets View	• Check that the DbSync service is running. Check the configuration, ensuring that the network addresses are correct. You may need to ensure that the ManagerAdapter on the site that you are syncing is configured and running. • If the configuration is correct and problems persist, it may be advisable to initiate a resync of that site through DIVA Connect's Admin utility.
A Copy, Delete, Cancel operation in the UI is returning access denied.	• Ensure that the correct Workflow Profile is configured in the *WebDefaultWorkflowProfile* in the ClientAdapter. Ensure that the request is permitted in the list of valid Messages section of the Workflow Profile. • If you are using Access Rules, ensure that **WebConnect** appears in an Include rule.

A

Sample Configuration Files

ClientAdapter Sample Configuration

```
<config>
    <!-- TOP-LEVEL -->
    <LocalSiteName>diva1</LocalSiteName>
    <LogLevel>INFO</LogLevel>
    <MaxClientConnections>50</MaxClientConnections>
    <WebServicePort>9801</WebManagementPort>

    <!-- ADAPTER PORTS -->
    <AdapterPorts>
        <AdapterPort>
            <ListenPort>7100</ListenPort>
            <RoutingType>Direct</RoutingType>
            <Sitename>diva1</Sitename>
        </AdapterPort>
        <AdapterPort>
            <ListenPort>7101</ListenPort>
            <RoutingType>MultiDiva</RoutingType>
            <WorkflowProfile>default</WorkflowProfile>
        </AdapterPort>
    </AdapterPorts>

    <!-- DIVA MANAGERS -->
    <DivaManagers>
        <DivaManager>
            <Sitename>diva1</Sitename>
            <ConnectionType>Socket</ConnectionType>
            <Address>127.0.0.1</Address>
            <Port>9000</Port>
        </DivaManager>
        <DivaManager>
            <Sitename>diva2</Sitename>
            <ConnectionType>WebService</ConnectionType>
            <BaseUrl>https://172.16.3.59:8101</BaseUrl>
        </DivaManager>
    </DivaManagers>

    <!-- DIVA Connect DATABASE -->
    <DivaconnectDatabase>
        <Address>127.0.0.1</Address>
        <Port>1521</Port>
        <User></User>
        <Password></Password>
    </DivaconnectDatabase>

    <!-- WORKFLOW PROFILES -->
    <WorkflowProfile>
        <Name>default</Name>
        <RestoreRetryAttempts>3</RestoreRetryAttempts>
        <Messages>
            <Message>Copy</Message>
            <Message>Archive</Message>
            <Message>Restore</Message>
            <Message>Delete</Message>
            <Message>Cancel</Message>
            <Message>AllInfo</Message>
            <Message>PartialRestore</Message>
        </Messages>

        <!-- SITE TO SITE TRANSFER CONFIG -->
        <Mappings>
            <SiteToSiteTransfer>
            <!-IF transferring…-->
                <FromSitename>diva1</FromSitename>
                <ToSitename>diva2</ToSitename>
            <!-THEN use these…-->
                <FromSrcDest>viva</FromSrcDest>
                <Type>RestoreAndArchive</Type>
                <ToDefaultMedia>disk_001</ToDefaultMedia>
                <ToSrcDest>arch</ToSrcDest>
            </SiteToSiteTransfer>
            <SiteToSiteTransfer>
            <!-IF transferring…-->
                <FromSitename>diva2</FromSitename>
                <ToSitename>diva1</ToSitename>
            <!-THEN use these…-->
                <FromSrcDest>viva</FromSrcDest>
                <Type>RestoreAndArchive</Type>
                <ToDefaultMedia>disk_002</ToDefaultMedia>
                <ToSrcDest>arch</ToSrcDest>
            </SiteToSiteTransfer>
        </Mappings>
    </WorkflowProfile>
</config>
```

ManagerAdapter Sample Configuration

```
<config>
    <!-- TOP-LEVEL -->
    <LocalSiteName>diva1</LocalSiteName>
    <ManagerAddress>127.0.0.1</ManagerAddress>
    <ManagerPort>9000</ManagerPort>
    <LogLevel>INFO</LogLevel>
    <WebServicePort>9800</WebServicePort>

    <!-- Manager DATABASE -->
    <ManagerDatabase>
        <Address>127.0.0.1</Address>
        <Port>1521</Port>
        <User></User>
        <Password></Password>
    </ManagerDatabase>
</config>
```

DbSync Sample Configuration

```
<config>
    <!-- TOP-LEVEL -->
    <LocalSiteName>diva1</LocalSiteName>
    <LogLevel>INFO</LogLevel>
    <WebServicePort>9802</WebServicePort>

    <!-Sites to SYNC -->
    <DivaManagers>
        <DivaManager>
            <Sitename>diva1</Sitename>
            <BaseUrl>https://127.0.0.1:9800</BaseURL>
        </DivaManager>
        <DivaManager>
            <Sitename>diva2</Sitename>
            <BaseUrl>https://172.16.3.59:9800</BaseURL>
        </DivaManager>
    </DivaManagers>

    <!-- DIVA Connect DATABASE -->
    <DivaconnectDatabase>
        <Address>127.0.0.1</Address>
        <Port>1521</Port>
        <User></User>
        <Password></Password>
    </DivaconnectDatabase>
</config>
```

Glossary

AXF (*Archive Exchange Format*)

A vendor-neutral format, supported by DIVA Core, for storing content on various mediums (*including tape and disk*). It can store many files and directories in a single file (*container*) much like a ZIP file.

Content Request

System operations that handle archived content — Archive, Copy, Delete, Restore.

DIVA Connect Home Directory

The file system directory where DIVA Connect is installed. DIVA Connect can be installed in its own directory (*recommended*), or within the DIVA Core home directory.

DIVA Connect Object (*Asset*)

An archived object describes digital content in the archive system. A DIVA Connect object is a set of archive objects, archived at one or more DIVA Core sites, sharing the same name (*object name and category pair*). Each object on a particular site can have one or more copies (*object instances*).

DIVA Connect Site

A sitename uniquely identifies a DIVA Connect site, which is defined as exactly one DIVA Core installation and one or more DIVA Connect services.

DFM (*Drop Folder Monitor*)

The DIVA Core DFM service monitors content that is being written to disk in a folder. It then performs the operation assigned to that folder (*Archive, Restore, Copy, and so on*) to the content (*placed in the folder*) into a DIVA Core system after the content has been completely written top the folder location.

Intersite Copy

An operation that copies content from one DIVA site to another, archiving the content to the target site.

Intersite Transfer

An operation that moves content from one site to another, but does not archive the content to the target site.

Media

A name that represents a set of archive media (*disks or tapes*) in a DIVA system. There exists one media per copy of an asset.

MultiDiva Mode

When an API client is connected to DIVA Connect in MultiDiva mode, multiple DIVA Core sites appear to the client as one large archive system.

Site-to-Site Mapping

Describes how DIVA Connect will copy objects from one site to another.

Server

Contains information necessary to communicate with a connected server or disk. You transfer content to and from DIVA Core through these servers and disks. Examples are Video servers, FTP servers, and disk arrays.

SPM (*Storage Plan Manager*)

You can configure the DIVA Core SPM (*Storage Plan Manager*) service to automatically perform an operation when content arrives at a DIVA Core site (*such as restore or copy*).

Workflow Profile

A set of parameters that defines how incoming requests will be processed by DIVA Connect. Multiple profiles can be created and mapped to input ports.