

GT 4.2.0 DRS: System Administrator's Guide

DRAFT

GT 4.2.0 DRS: System Administrator's Guide

Introduction

This guide contains advanced configuration information for system administrators working with the Data Replication Service (DRS). It provides references to information on procedures typically performed by system administrators, including installing, configuring, deploying, and testing the installation.

Important

This information is in addition to the basic Globus Toolkit prerequisite, overview, installation, security configuration instructions in the [Installing GT 4.2.0](#). Read through this guide before continuing!

DRAFT

Table of Contents

1. Building and installing	1
1. Installing DRS using Globus Toolkit Binary Bundle	1
2. Installing DRS using Globus Toolkit Source Bundle	1
3. Installing DRS using Globus CVS	1
2. Configuring	3
1. Configuration overview	3
2. Syntax of the interface	3
3. Testing	4
4. Security considerations	5
1. Data Replication Service (DRS) Security Considerations	5
5. Debugging	6
1. Verbose log output	6
2. Logging in Java WS Core	6
6. Troubleshooting	8
1. Errors	8
Index	9

List of Tables

6.1. Data Replication Service (DRS) Errors 8

DRAFT

Chapter 1. Building and installing

DRS is distributed with the Globus Toolkit 4.2.0 and is available in both the binary bundles and the source bundle. For most purposes, the binary bundle provides the simplest means of installing the DRS and its requirements. There are three typical ways to access and install the DRS:

- [Section 1, “Installing DRS using Globus Toolkit Binary Bundle”](#)
- [Section 2, “Installing DRS using Globus Toolkit Source Bundle”](#)
- [Section 3, “Installing DRS using Globus CVS”](#)

Important

Before proceeding with installation of DRS, please familiarize yourself with the installation procedures for RLS (see [Replica Location Service \(RLS\)](#)) and RFT (see [Reliable File Transfer \(RFT\) Service](#)).

1. Installing DRS using Globus Toolkit Binary Bundle

Follow instructions provided by [Installing GT 4.2.0](#) and [Installing GT](#) in order to install the DRS from one of the available binary bundles of the Globus Toolkit.

Once you have unpackaged the binary bundle, the following commands may be used to install DRS:

```
% ./configure --prefix=$GLOBUS_LOCATION --enable-rls --enable-drs
% make
% make install
```

2. Installing DRS using Globus Toolkit Source Bundle

In order to install DRS, you will need to ensure that RLS is also installed as part of the build by reviewing its installation procedures. The following commands will install the DRS (optionally you may use the "drs" make target to build only DRS and its requirements:

```
% ./configure --prefix=/path/to/install --enable-rls --enable-drs
% make [drs]
% make install
```

3. Installing DRS using Globus CVS

Additionally, you may access the DRS from the Globus Toolkit CVS repository. You may deploy the 4.0.x DRS on top of an existing GT 4.0.x installation using the `globus_4_0_branch` tag. You may deploy the 4.1.x DRS on top of an existing GT 4.2 installation using the `globus_4_2_branch` tag.

The following example instructions will only work given the presence of an existing GT installation with *all DRS dependencies*. The best way to establish such an environment is to first follow the basic installation instructions and subsequently use the following instructions to obtain the latest updates from the CVS branch. Note the use of the `globus_X_Y_branch` tag depends on whether you are interested in updates from the maintenance branch or the development trunk.

```
% setenv GLOBUS_LOCATION /path/to/existing/gtXY/install/  
% setenv CVSROOT :pserver:anonymous@cvs.globus.org:/home/globdev/CVS/globus-packages  
% cvs co [-r globus_X_Y_branch] ws-replica/replicator/  
% cd ws-replica/replicator/  
% ant deploy
```

To learn more about general instructions regarding GT CVS access, see [Remote CVS Access](#)¹.

¹ <http://www.globus.org/toolkit/docs/development/remote-cvs.html>

Chapter 2. Configuring

This information is in addition to the basic configuration instructions in the [Installing GT 4.2.0](#). Aside from the basic configuration of GT 4.2.0, please review the following instructions:

1. Configuration overview

The DRS requires certain JNDI settings to be properly configured. The installed JNDI configuration file may be found at `$GLOBUS_LOCATION/etc/globus_wsrp_replicator/jndi-config.xml`. To view the default configuration file (shipped with the GT 4.2.0 release) from the Globus CVS repository [click here](#)¹.

2. Syntax of the interface

The settings are structured as name-value pairs. For example:

```
<parameter>
  <name>defaultIndexUrl</name>
  <value>rls://127.0.0.1:39281</value>
</parameter>
```

The following settings must be properly configured:

- `proxyfileDir`: the directory that you would like the DRS to temporarily store user proxies. No setting is necessary. This value may be empty.
- `requestfileDir`: the directory that you would like the DRS to temporarily store request files. No setting is necessary. This value may be empty.
- `defaultIndexUrl`: the connection URL for your installation of RLS running as a RLI service.
- `defaultRegistrationUrl`: the connection URL for your installation of RLS running as a LRC service.
- `defaultReliableTransferUrl`: the connection URL for your installation of the RFT ReliableFileTransferFactoryService.
- `proxyfileChangePermsCmd`: the platform-dependent command to change file permissions to user-only read-write permissions.
- The rest of the parameter/value pairs may retain the given default values.

¹ http://viewcvs.globus.org/viewcvs.cgi/ws-replica/replicator/service/java/source/deploy-jndi-config.xml?tab=globus_4_2_branch&content-type=text/vnd.viewcvs-markup

Chapter 3. Testing

This service does not provide a set of tests yet.

DRAFT

Chapter 4. Security considerations

1. Data Replication Service (DRS) Security Considerations

1.1. Service configuration files

The service configuration files such as the JNDI configuration file, `jndi-config.xml`, and the Web service deployment descriptor, `server-config.wsdd`, located in the `$GLOBUS_LOCATION/etc/globus_wsrf_replicator` directory, contain sensitive information such as database username and password. It is important to ensure that these files are readable only by the system administrator that is responsible for the container. During deployment, the permissions on these files are adjusted automatically, however, you should verify the permissions to ensure that they have been correctly set for your specific platform.

1.2. Delegated proxy credential files

Creating a Replicator requires that the user supply a delegated credential to the DRS during the initial creation request. The service retrieves the delegated credential from the Delegation Service and stores it on the file system. As part of the DRS configuration (see installation and configuration instructions), the user selects a directory to use for storage of delegated credentials. The default setting is for the DRS to store the file in the system's designated temporary directory (e.g., `/tmp` on many platforms). The service sets the permissions on the temporary file such that it can only be accessed by the user account used to run the container.

Chapter 5. Debugging

1. Verbose log output

Generating verbose log output is a critical aid in troubleshooting of the DRS and is useful when communicating problems to Globus support lists. To increase logging detail, add the following line to the `$GLOBUS_LOCATION/container-log4j.properties` file.

```
...
log4j.category.org.globus.replica=DEBUG
...
```

2. Logging in Java WS Core

The following information applies to Java WS Core and all services built on Java WS Core.

Java WS Core server side has two types of loggers. One logger is used for development logging and by default writes to standard out. The other logger includes system administration information and is [CEDPs best practices](#)¹ compliant.

On client side, only developer logging is available and is configured using `log4j.properties`.

2.1. Development Logging in Java WS Core

The following information applies to Java WS Core and those services built on it.

Logging in the Java WS Core is based on the [Jakarta Commons Logging](#)² API. Commons Logging provides a consistent interface for instrumenting source code while at the same time allowing the user to plug-in a different logging implementation. Currently we use [Log4j](#)³ as a logging implementation. Log4j uses a separate configuration file to configure itself. Please see Log4j documentation for details on the [configuration file format](#)⁴.

2.1.1. Configuring server side developer logs

Server side logging can be configured in `$GLOBUS_LOCATION/container-log4j.properties`, when the container is stand alone container. For tomcat level logging, refer to [Logging for Tomcat](#)⁵. The logger `log4j.appender.A1` is used for developer logging and by default writes output to the system output. By default it is set for all warnings in the Globus Toolkit package to be displayed.

Additional logging can be enabled for a package by adding a new line to the configuration file. Example:

```
#for debug level logging from org.globus.package.FooClass
log4j.category.org.globus.package.name.FooClass=DEBUG
#for warnings from org.some.warn.package
```

¹ <http://cedps.net/index.php/LoggingBestPractices>

² <http://jakarta.apache.org/commons/logging/>

³ <http://logging.apache.org/log4j/>

⁴ [http://logging.apache.org/log4j/docs/api/org/apache/log4j/PropertyConfigurator.html#doConfigure\(java.lang.String,org.apache.log4j.spi.LoggerRepository\)](http://logging.apache.org/log4j/docs/api/org/apache/log4j/PropertyConfigurator.html#doConfigure(java.lang.String,org.apache.log4j.spi.LoggerRepository))

⁵ <http://tomcat.apache.org/tomcat-5.5-doc/logging.html>

```
log4j.category.org.some.warn.package=WARN
```

2.1.2. Configuring client side developer logs

Client side logging can be configured in `$GLOBUS_LOCATION/log4j.properties`. The logger `log4j.appender.A1` is used for developer logging and by default writes output to the system output. By default it is set for all warnings in the Globus Toolkit package to be displayed.

2.2. Configuring system administration logs

The specific logger to edit will be `log4j.logger.sysadmin` in `$GLOBUS_LOCATION/container-log4j.properties`. There you can configure the following properties:

```
log4j.appender.infoCategory=org.apache.log4j.RollingFileAppender
log4j.appender.infoCategory.Threshold=INFO
log4j.appender.infoCategory.File=var/containerLog
log4j.appender.infoCategory.MaxFileSize=10MB
log4j.appender.infoCategory.MaxBackupIndex=2
```

Above implies the logging file is rolling with each file size limited to 10MB and the logging information is stored in `$GLOBUS_LOCATION/var/containerLog`.

2.3. Sample log file

The [sample log file](#)⁶ contains many log entries for various scenarios in the Java WS container.

⁶ <http://www.globus.org/toolkit/docs/4.2/4.2.0/common/javawscore/sample-container-log.txt>

Chapter 6. Troubleshooting

For a list of common errors in GT, see [Error Codes](#).

1. Errors

Table 6.1. Data Replication Service (DRS) Errors

Error Code	Definition	Possible Solutions
Authorization failed. Expected <hostname1> target but received <hostname2>	Did not receive expected hostname	When authorization is enabled on the container, you may need to use the proper hostname when referencing the DRS service rather than using localhost.
org.globus.wsrp.ResourceException: Failed to create Replication: /scratch/testrun (No such file or directory)	Cannot find the request file	Ensure that the request file's filename is correct, that it is reachable by the DRS service, and that it has the appropriate permissions for the DRS service to access it.
org.globus.wsrp.ResourceException: Failed to create Replication: String index out of range: -1	The request file is malformed (for example by using spaces instead of a delimiting tab character) which is resulting in a runtime exception.	Make sure your request file is in the correct form as described here .

Index

D

debugging
 logging, 6

E

errors, 8

I

installing
 using binary, 1
 using CVS, 1
 using source, 1

L

logging, 6
 CEDPS-compliant, 6
 debugging, 6

DRAFT

GT 4.2.0 DRS: User's Guide

DRAFT

GT 4.2.0 DRS: User's Guide

Introduction

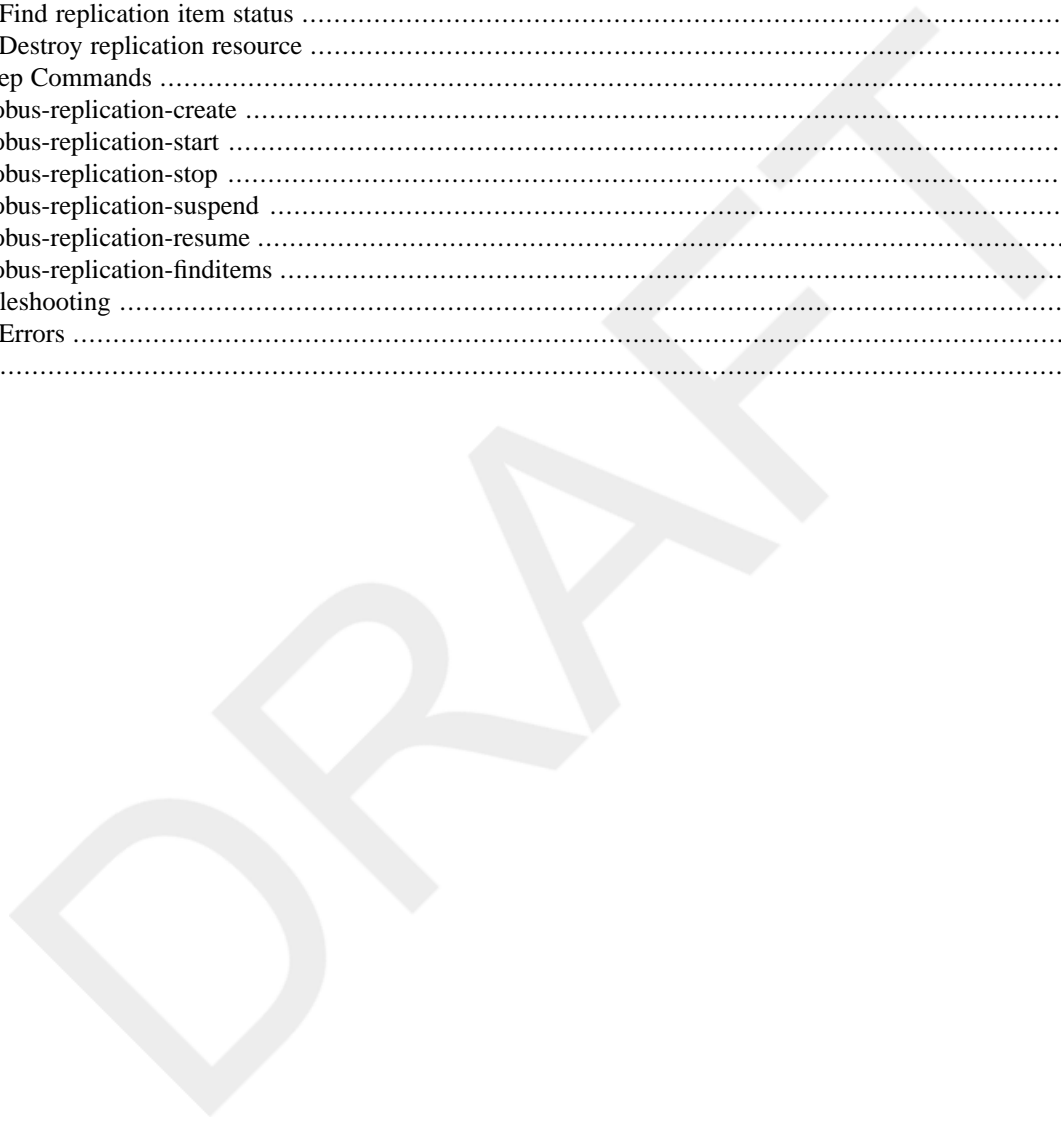
DRS is intended to provide a general utility for replicating data files by locating physical file names in the Replica Location Service (RLS), transferring the data files using the Reliable File Transfer (RFT) service, and then registering the new physical file locations in the RLS again. In order to use the DRS, the RLS and RFT must also be installed. In order for DRS to locate your files, you must first register them in the RLS. Once you have registered your files you may use the DRS to replicate them to locations throughout your data Grid.

The DRS provides a set of command-line tools to create replication requests, to start/suspend/resume/stop replications, and to find status information on replications. The usage scenarios in this guide provide examples for basic usage of the DRS.

DRAFT

Table of Contents

- 1. Managing Sets of Replicas (WS) 1
 - 1. Generate a valid proxy 1
 - 2. Delegate user credentials 1
 - 3. Replication request file 1
 - 4. Create replication resource 2
 - 5. Start replication 2
 - 6. Get replication resource properties 2
 - 7. Find replication item status 3
 - 8. Destroy replication resource 4
- I. DataRep Commands ?
 - globus-replication-create 6
 - globus-replication-start 8
 - globus-replication-stop 9
 - globus-replication-suspend 10
 - globus-replication-resume 11
 - globus-replication-finditems 12
- 2. Troubleshooting 13
 - 1. Errors 13
- Index 14



List of Tables

1. Options	7
2. Options	8
3. Options	9
4. Options	10
5. Options	11
6. Options	12
2.1. Data Replication Service (DRS) Errors	13

DRAFT

Chapter 1. Managing Sets of Replicas (WS)

This section describes a few key usage scenarios and provides examples of using the DRS command-line tools.

1. Generate a valid proxy

Before using any of the tools, a user must generate a valid user proxy. Use `grid-proxy-init`.

```
% $GLOBUS_LOCATION/bin/grid-proxy-init
Your identity: /O=Grid/OU=GlobusTest/OU=simpleCA.mymachine/OU=mymachine/CN=John Doe
Enter GRID pass phrase for this identity:
Creating proxy ..... Done
Your proxy is valid until: Tue Oct 26 01:33:42 2004
```

2. Delegate user credentials

Once you have generated a valid proxy you must create a delegated credential. Your delegated credential will be used by the DRS to make secure calls to other services (e.g., RLS, RFT, etc.) in order to perform the data replication. It is important to ensure that you give your delegated credential enough lifetime to support the running time of your replication activities. To delegate your credential use `globus-credential-delegate(1)`.

```
% $GLOBUS_LOCATION/bin/globus-credential-delegate -s https://localhost:8443/wsrf/services/
Delegated credential EPR:
Address: https://localhost:8443/wsrf/services/DelegationService
Reference property[0]:
<nsl:DelegationKey xmlns:nsl="http://www.globus.org/08/2004/delegationService">b90c6c90-fc
```

3. Replication request file

A key parameter for any replication request is the request file. The replication request file is a text file containing CRLF-terminated rows of tab-delimited pairs of Logical Filename (LFN) names and destination (URL) locations. An example of such a file is shown.

```
% cat testrun.req
testrun-1      gsiftp://myhost:9001/sandbox/files/testrun-1
testrun-2      gsiftp://myhost:9001/sandbox/files/testrun-2
testrun-3      gsiftp://myhost:9001/sandbox/files/testrun-3
testrun-4      gsiftp://myhost:9001/sandbox/files/testrun-4
testrun-5      gsiftp://myhost:9001/sandbox/files/testrun-5
```

 **Note**

The LFNs in the left column of the request file (e.g., `testrun-1`, `testrun-2`, and so on shown in the example) must be registered in the RLS catalog and indexed in the RLS index service. This typically involves using the `add RLI` command (e.g., `globus-rls-admin -a rls-receiver-url rls-sender-url`) to tell the RLS to send updates to another (or the same) RLS, and then the `create` command (e.g., `globus-rls-cli create testrun-1 gsiftp://sourcehost:9001/path/to/testrun-1 rls-sender-url`) to register the LFN at the RLS catalog service. For details see [globus-rls-admin\(1\)](#) and [globus-rls-cli\(1\)](#).

4. Create replication resource

The initial step for any replication is to create the replication resource. Creating the resource depends on the availability of a DRS service, a delegated credential, and a properly formatted replication request file. The replication request file must be specified by its URL. Currently supported URL schemes for the request file include `file`, `http`, and `ftp`. If the replication client is run local to the service the `file` scheme is appropriate, whereas if the client is remote than the latter schemes must be used. It is a good practice to specify a filename to save the replication resource's endpoint reference. The endpoint reference is required for all other operations on the resource, such as getting resource properties, starting/stopping, and destroying it. Numerous options are available to influence the behavior of the data replication activities (see [globus-replication-create\(1\)](#)). One option of particular interest is the `--start` option, which immediately starts the replication activities following creation of the replication resource. An example of using the [globus-replication-create\(1\)](#) tool is shown.

```
% $GLOBUS_LOCATION/bin/globus-replication-create -s \  
  https://myhost:8443/wsrf/services/ReplicationService \  
  -C mycredential.epr -V myreplicator.epr file:///scratch/testrun.req
```

This command does not write to `stdout` when successful unless the `--debug` option is specified.

5. Start replication

Once a replication resource has been create, the replication activities may be started. As mentioned in [Create replication resource](#) the replication may be immediately started after it is created. If the immediate start option is not specified, the [globus-replication-start\(1\)](#) tool must be used to start the replication.

```
% $GLOBUS_LOCATION/bin/globus-replication-start -e myreplicator.epr
```

No output is expect from this command when successful.

6. Get replication resource properties

Throughout the lifecycle and after the completion of the replication resource, it will be important to lookup its Resource Properties. The standard WS-RF port types are supported and the supplied tools (e.g., [wsrf-get-property](#)) may be used with the DRS and its resources.

```
% $GLOBUS_LOCATION/bin/wsrf-get-property -e myreplicator.epr \  
  "{http://www.globus.org/namespaces/2005/05/replica/replicator}status"
```

```
<ns1:status xmlns:ns1="http://www.globus.org/namespaces/2005/05/replica/replicator">
Active</ns1:status>
```

And,

```
% $GLOBUS_LOCATION/bin/wsrfe-get-property -e myreplicator.epr \
  "{http://www.globus.org/namespaces/2005/05/replica/replicator}count"
<ns1:count xmlns:ns1="http://www.globus.org/namespaces/2005/05/replica/replicator">
  <ns1:total>10</ns1:total>
  <ns1:finished>0</ns1:finished>
  <ns1:failed>0</ns1:failed>
  <ns1:terminated>0</ns1:terminated>
</ns1:count>
```

7. Find replication item status

Throughout the lifecycle and after the completion of the replication resource, it may be helpful to find individual replication items in the replication resource to inspect the detailed status of the replication activities. The `globus-replication-finditems(1)` tool is used to find replication items. The following example demonstrates the usage when finding a limited number of items, offset into the lookup results set, for a specified status.

```
% $GLOBUS_LOCATION/bin/globus-replication-finditems -e myreplicator.epr -S Pending -O 1 -L
<ns1:FindItemsResponse xmlns:ns1="http://www.globus.org/namespaces/2005/05/replica/replica"
  <ns1:items xsi:type="ns1:ReplicationItemType" xmlns:xsi="http://www.w3.org/2001/XMLSchema"
    <ns1:uri xsi:type="xsd:string" xmlns:xsd="http://www.w3.org/2001/XMLSchema">testrun-2</ns1:uri>
    <ns1:priority xsi:type="xsd:int" xmlns:xsd="http://www.w3.org/2001/XMLSchema">1000</ns1:priority>
    <ns1:status xsi:type="ns1:ReplicationItemStatusEnumerationType">Pending</ns1:status>
    <ns1:destinations xsi:type="ns1:DestinationType">
      <ns1:uri xsi:type="xsd:string" xmlns:xsd="http://www.w3.org/2001/XMLSchema">
        gsiftp://myhost:9001/sandbox/files/testrun-2</ns1:uri>
      <ns1:status xsi:type="ns1:DestinationStatusEnumerationType">Pending</ns1:status>
    </ns1:destinations>
  </ns1:items>
  <ns1:items xsi:type="ns1:ReplicationItemType" xmlns:xsi="http://www.w3.org/2001/XMLSchema"
    <ns1:uri xsi:type="xsd:string" xmlns:xsd="http://www.w3.org/2001/XMLSchema">testrun-3</ns1:uri>
    <ns1:priority xsi:type="xsd:int" xmlns:xsd="http://www.w3.org/2001/XMLSchema">1000</ns1:priority>
    <ns1:status xsi:type="ns1:ReplicationItemStatusEnumerationType">Pending</ns1:status>
    <ns1:destinations xsi:type="ns1:DestinationType">
      <ns1:uri xsi:type="xsd:string" xmlns:xsd="http://www.w3.org/2001/XMLSchema">
        gsiftp://myhost:9001/sandbox/files/testrun-3</ns1:uri>
      <ns1:status xsi:type="ns1:DestinationStatusEnumerationType">Pending</ns1:status>
    </ns1:destinations>
  </ns1:items>
</ns1:FindItemsResponse>
```

8. Destroy replication resource

When the replication is complete, the replication resource may be destroyed. Destroying the replication resource will help to free up system resources (namely, memory), especially in the case that the replication entailed a large amount of individual replication activities (i.e., many files specified in the replication request file). The standard WS-RF port types are supported and the supplied `wrsf-destroy` tool may be used to destroy the DRS resource.

```
% $GLOBUS_LOCATION/bin/wrsf-destroy -e myreplicator.epr  
Destroy operation was successful
```

DRAFT

DataRep Commands

The DRS provides a set of command-line tools to control the creation and lifecycle of a given replication request. These command line tools are available on Unix and Windows platforms and will work in the same way (of course within the platform rules - the path syntax, variable definitions, etc.).

DRAFT

Name

`globus-replication-create` -- This tool is used to create a replication resource by submitting a replication request to the designated replication service.

```
globus-replication-create
```

Tool description

Use this tool to create replication resources (also referred to as "Replicator" resources). You must specify the URL of the ReplicationService where the resource will be created. You must submit the filename of a file containing an Endpoint Reference (EPR) to a delegated credential resource, which you must have previously created. Finally, you must submit the URL of a request file specifying the desired data replications. If the client is running local to the service container the URL may be a `file://` URL, whereas if the client is remote the URL may be a `http://` or `ftp://` URL. The request file adopts a table format structure where each line in the file represents a source-destination pair delimited by a single *tab* character. The source should be a logical filename (LFN) as found in a Replica Location Service (RLS) Replica Location Index (RLI) service. The destination should be a URL acceptable to the GridFTP server. Most likely, you will want to specify a filename in order to save the newly created Replicator resource's EPR. You may use the EPR for starting the resource and querying its resource properties.

Command syntax

```
globus-replication-create [options] request-file
```

Table 1. Options

-a,--anonymous	Use anonymous authentication. (requires either -m 'conv' or transport (https) security)
--binary <boolean>	Specifies binary data transfer
--blockSize <int>	Block size for data transfer
-c,--serverCertificate <file>	A file with server's certificate used for encryption. Used in the case of GSI Secure Message encryption
-C,--delegatedCredential <file>	Loads Delegated Credential EPR from file
--concurrency <int>	Concurrency of data transfer
-d,--debug	Enables debug mode
--dataChannelAuth <boolean>	Data channel authentication for transfers
--destinationSubject <name>	Destination subject name for data transfer
-e,--eprFile <file>	Loads EPR from file
-f,--descriptor <file>	Sets client security descriptor. Overrides all other security settings
-g,--delegation <mode>	Performs delegation. Can be 'limited' or 'full'. (requires -m 'conv')
-h,--help	Displays help
-k,--key <name value>	Resource Key
-l,--contextLifetime <value>	Lifetime of context created for GSI Secure Conversation (requires -m 'conv')
-m,--securityMech <type>	Sets authentication mechanism: 'msg' (for GSI Secure Message), or 'conv' (for GSI Secure Conversation)
-p,--protection <type>	Sets protection level, can be 'sig' (for signature) can be 'enc' (for encryption)
--parallelStreams <int>	Parallel streams for data transfer
-s,--service <url>	Service URL
-S,--start	Starts the Replicator resource immediately
--sourceSubject <name>	Source subject name for data transfer
--subject <name>	Subject name for data transfer
--tcpBufferSize <int>	TCP buffer size for data transfer
--userName <name>	User name for data transfer
-V,--saveEpr <file>	Save EPR of newly created Replicator to file
-z,--authorization <type>	Sets authorization, can be 'self', 'host' or 'none'

Name

globus-replication-start -- This tool starts the replication activities.

globus-replication-start

Tool description

Replication resources created with the `globus-replication-create` tool may be "started" by using this tool and passing the filename of the saved EPR as a parameter to the tool. The tool will indicate an error condition if the user attempts to start a resource that has been previously started.

Command syntax

```
globus-replication-start [options]
```

Table 2. Options

-a,--anonymous	Use anonymous authentication. (requires either -m 'conv' or transport (https) security)
-c,--serverCertificate <file>	A file with server's certificate used for encryption. Used in the case of GSI Secure Message encryption
-d,--debug	Enables debug mode
-e,--eprFile <file>	Loads EPR from file
-f,--descriptor <file>	Sets client security descriptor. Overrides all other security settings
-g,--delegation <mode>	Performs delegation. Can be 'limited' or 'full'. (requires -m 'conv')
-h,--help	Displays help
-k,--key <name value>	Resource Key
-l,--contextLifetime <value>	Lifetime of context created for GSI Secure Conversation (requires -m 'conv')
-m,--securityMech <type>	Sets authentication mechanism: 'msg' (for GSI Secure Message), or 'conv' (for GSI Secure Conversation)
-p,--protection <type>	Sets protection level, can be 'sig' (for signature) can be 'enc' (for encryption)
-s,--service <url>	Service URL
-z,--authorization <type>	Sets authorization, can be 'self', 'host' or 'none'

Name

globus-replication-stop -- This tool stops the replication activities.

globus-replication-stop

Tool description

Replication resources created with the `globus-replication-create` tool may be "stoped" by using this tool and passing the filename of the saved EPR as a parameter to the tool. The tool will indicate an error condition if the user attempts to stop a resource that has not been previously started, a resource that has been suspended, or a resource that has terminated or been destroyed.

Command syntax

```
globus-replication-stop [options]
```

Table 3. Options

-a,--anonymous	Use anonymous authentication. (requires either -m 'conv' or transport (https) security)
-c,--serverCertificate <file>	A file with server's certificate used for encryption. Used in the case of GSI Secure Message encryption
-d,--debug	Enables debug mode
-e,--eprFile <file>	Loads EPR from file
-f,--descriptor <file>	Sets client security descriptor. Overrides all other security settings
-g,--delegation <mode>	Performs delegation. Can be 'limited' or 'full'. (requires -m 'conv')
-h,--help	Displays help
-k,--key <name value>	Resource Key
-l,--contextLifetime <value>	Lifetime of context created for GSI Secure Conversation (requires -m 'conv')
-m,--securityMech <type>	Sets authentication mechanism: 'msg' (for GSI Secure Message), or 'conv' (for GSI Secure Conversation)
-p,--protection <type>	Sets protection level, can be 'sig' (for signature) can be 'enc' (for encryption)
-s,--service <url>	Service URL
-z,--authorization <type>	Sets authorization, can be 'self', 'host' or 'none'

Name

globus-replication-suspend -- This tool suspends the replication activities.

globus-replication-suspend

Tool description

Replication resources created with the `globus-replication-create` tool may be "suspended" by using this tool and passing the filename of the saved EPR as a parameter to the tool. The tool will indicate an error condition if the user attempts to suspend a resource that has not been previously started, a resource that has been suspended, or a resources that is done or has been destroyed.

Command syntax

```
globus-replication-suspend [options]
```

Table 4. Options

-a,--anonymous	Use anonymous authentication. (requires either -m 'conv' or transport (https) security)
-c,--serverCertificate <file>	A file with server's certificate used for encryption. Used in the case of GSI Secure Message encryption
-d,--debug	Enables debug mode
-e,--eprFile <file>	Loads EPR from file
-f,--descriptor <file>	Sets client security descriptor. Overrides all other security settings
-g,--delegation <mode>	Performs delegation. Can be 'limited' or 'full'. (requires -m 'conv')
-h,--help	Displays help
-k,--key <name value>	Resource Key
-l,--contextLifetime <value>	Lifetime of context created for GSI Secure Conversation (requires -m 'conv')
-m,--securityMech <type>	Sets authentication mechanism: 'msg' (for GSI Secure Message), or 'conv' (for GSI Secure Conversation)
-p,--protection <type>	Sets protection level, can be 'sig' (for signature) can be 'enc' (for encryption)
-s,--service <url>	Service URL
-z,--authorization <type>	Sets authorization, can be 'self', 'host' or 'none'

Name

globus-replication-resume -- This tool resumes the replication activities.

globus-replication-resume

Tool description

Replication resources created with the `globus-replication-create` tool may be "resumed" by using this tool and passing the filename of the saved EPR as a parameter to the tool. The tool will indicate an error condition if the user attempts to resume a resource that has not been previously suspended, or a resource that is done or has been destroyed.

Command syntax

```
globus-replication-resume [options]
```

Table 5. Options

-a,--anonymous	Use anonymous authentication. (requires either -m 'conv' or transport (https) security)
-c,--serverCertificate <file>	A file with server's certificate used for encryption. Used in the case of GSI Secure Message encryption
-d,--debug	Enables debug mode
-e,--eprFile <file>	Loads EPR from file
-f,--descriptor <file>	Sets client security descriptor. Overrides all other security settings
-g,--delegation <mode>	Performs delegation. Can be 'limited' or 'full'. (requires -m 'conv')
-h,--help	Displays help
-k,--key <name value>	Resource Key
-l,--contextLifetime <value>	Lifetime of context created for GSI Secure Conversation (requires -m 'conv')
-m,--securityMech <type>	Sets authentication mechanism: 'msg' (for GSI Secure Message), or 'conv' (for GSI Secure Conversation)
-p,--protection <type>	Sets protection level, can be 'sig' (for signature) can be 'enc' (for encryption)
-s,--service <url>	Service URL
-z,--authorization <type>	Sets authorization, can be 'self', 'host' or 'none'

Name

globus-replication-finditems -- This tool queries the replication resource to return the status of individual replication item activities.

globus-replication-finditems

Tool description

This tool provides the ability to query the status of individual replication items (e.g., replication of a specific file or files) managed by the given Replication resources. It is possible to query for the status of a specific named item or to query for the status of multiple items based on a particular status (e.g., Pending, Finished, Failed). In addition, to reduce potentially large overhead of returning a large results set to the client, the client may specify an offset and limit for the results set to be returned. The "name" or "status" option must be specified.

Command syntax

```
globus-replication-finditems [options] {-N name | -S status}
```

Table 6. Options

-a,--anonymous	Use anonymous authentication. (requires either -m 'conv' or transport (https) security)
-c,--serverCertificate <file>	A file with server's certificate used for encryption. Used in the case of GSI Secure Message encryption
-d,--debug	Enables debug mode
-e,--eprFile <file>	Loads EPR from file
-f,--descriptor <file>	Sets client security descriptor. Overrides all other security settings
-g,--delegation <mode>	Performs delegation. Can be 'limited' or 'full'. (requires -m 'conv')
-h,--help	Displays help
-k,--key <name value>	Resource Key
-l,--contextLifetime <value>	Lifetime of context created for GSI Secure Conversation (requires -m 'conv')
-L,--limit <num>	Limit on the size of the result set.
-m,--securityMech <type>	Sets authentication mechanism: 'msg' (for GSI Secure Message), or 'conv' (for GSI Secure Conversation)
-N,--byName <name>	Finds item by the Logical Filename (LFN) name.
-O,--offset <num>	Offset into the results set. Indexed by 0.
-p,--protection <type>	Sets protection level, can be 'sig' (for signature) can be 'enc' (for encryption)
-S,--byStatus <status>	Finds item(s) by status. Valid status values include "Pending", "Finished", "Failed", and "Terminated".
-s,--service <url>	Service URL
-z,--authorization <type>	Sets authorization, can be 'self', 'host' or 'none'

Chapter 2. Troubleshooting

The following section provides information about common errors for end users. For a list of common errors in GT, see [Error Codes](#). You can also find information on sys admin logs in [Chapter 5, Debugging](#).

1. Errors

Table 2.1. Data Replication Service (DRS) Errors

Error Code	Definition	Possible Solutions
Authorization failed. Expected <hostname1> target but received <hostname2>	Did not receive expected hostname	When authorization is enabled on the container, you may need to use the proper hostname when referencing the DRS service rather than using localhost.
org.globus.wsrp.ResourceException: Failed to create Replication: /scratch/testrun (No such file or directory)	Cannot find the request file	Ensure that the request file's filename is correct, that it is reachable by the DRS service, and that it has the appropriate permissions for the DRS service to access it.
org.globus.wsrp.ResourceException: Failed to create Replication: String index out of range: -1	The request file is malformed (for example by using spaces instead of a delimiting tab character) which is resulting in a runtime exception.	Make sure your request file is in the correct form as described here .

Index

D

delegating user credentials, 1

E

errors, 13

G

generating a valid proxy, 1

R

replica sets, 1

replication activities

 resuming, 11

 stopping, 9

 suspending, 10

replication request file, 1

replication resource

 creating, 2, 6

 destroying, 4

 finding replication item status, 3, 12

 getting replication resource properties, 2

 starting, 2, 8

DRAFT

GT 4.2.0 DRS: Developer's Guide

DRAFT

GT 4.2.0 DRS: Developer's Guide

Introduction

This guide contains information of interest to developers working with DRS. It provides reference information for application developers, including APIs, architecture, procedures for using the APIs and code samples.

DRAFT

Table of Contents

1. Before you begin	1
1. Feature summary	1
2. Tested platforms	1
3. Backward compatibility summary	1
4. Technology dependencies	2
5. Data Replication Service (DRS) Security Considerations	2
2. Usage scenarios	3
3. Tutorials	4
4. Architecture and design overview	5
5. APIs	6
1. Programming Model Overview	6
2. Component API	6
6. Services and WSDL	7
1. Protocol overview	7
2. Operations	7
3. Data Replication Service (DRS) Resource properties	8
4. Faults	8
5. WSDL and Schema Definition	9
I. DataRep Commands	?
globus-replication-create	11
globus-replication-start	13
globus-replication-stop	14
globus-replication-suspend	15
globus-replication-resume	16
globus-replication-finditems	17
7. Replication request file	18
1. Interface introduction	18
2. Syntax of the interface	18
8. Configuring	19
1. Configuration overview	19
2. Syntax of the interface	19
9. Environment variable interface	20
1. Environmental variables for DRS	20
10. Debugging	21
1. Development Logging in Java WS Core	21
2. Enable Debug Logging for DRS	21
11. Troubleshooting	22
1. Errors	22
12. Related Documentation	23
Index	24

List of Tables

1. Options	12
2. Options	13
3. Options	14
4. Options	15
5. Options	16
6. Options	17
11.1. Data Replication Service (DRS) Errors	22

DRAFT

Chapter 1. Before you begin

1. Feature summary

Features new in release GT 4.2.0:

- None.

Other Supported Features

- Improved implementation of the Data Replication Service: a WS-Resource, called the *Replicator*, which accepts a request from a client to locate, transfer, and register new replicas of data files in the Grid environment.
- A set of command-line tools to create (`globus-replication-create`), start (`globus-replication-start`), stop (`globus-replication-stop`), suspend (`globus-replication-suspend`), resume (`globus-replication-resume`) replication requests, and find item status (`globus-replication-finditems`).
- WSDL-defined SOAP operations to *create*, *start*, *stop*, *suspend*, and *resume* a replication request, along with operations to get the status of individual replicas in the request. For details, [click here](#)¹ to view a listing of the WSDL-defined interface from the Globus CVS repository.
- APIs to allow users to implement custom replica source selection algorithms.
- Supports secure transport, secure conversation, and secure message communication as provided by GT 4.2.0.

Deprecated Features

- Database-backed State Persistence: State is now maintained in memory and lasts only for the lifetime of the WS-Resource or as dictated by the service container. This change simplifies setup of the DRS. We intend to reintroduce other persistence model(s) after we have collected additional user feedback on the DRS.

2. Tested platforms

Tested Platforms for DRS

- Linux (RedHat, Debian)

3. Backward compatibility summary

Protocol changes since GT version 4.0.x:

- None

API changes since GT version 4.0.x:

- None

Exception changes since GT version 4.0.x:

¹ http://viewcvs.globus.org/viewcvs.cgi/ws-replica/replicator/common/schema/replica/replicator/?only_with_tag=globus_4_0_branch

- None

Schema changes since GT version 4.0.x:

- None

4. Technology dependencies

DRS depends on the following GT components:

- Java WS Core
- WS Authentication and Authorization
- Delegation Service
- RFT
- RLS

DRS depends on the following 3rd party software:

- None

5. Data Replication Service (DRS) Security Considerations

5.1. Service configuration files

The service configuration files such as the JNDI configuration file, `jndi-config.xml`, and the Web service deployment descriptor, `server-config.wsdd`, located in the `$GLOBUS_LOCATION/etc/globus_wsrf_replicator` directory, contain sensitive information such as database username and password. It is important to ensure that these files are readable only by the system administrator that is responsible for the container. During deployment, the permissions on these files are adjusted automatically, however, you should verify the permissions to ensure that they have been correctly set for your specific platform.

5.2. Delegated proxy credential files

Creating a Replicator requires that the user supply a delegated credential to the DRS during the initial creation request. The service retrieves the delegated credential from the Delegation Service and stores it on the file system. As part of the DRS configuration (see installation and configuration instructions), the user selects a directory to use for storage of delegated credentials. The default setting is for the DRS to store the file in the system's designated temporary directory (e.g., `/tmp` on many platforms). The service sets the permissions on the temporary file such that it can only be accessed by the user account used to run the container.

Chapter 2. Usage scenarios

Not available.

DRAFT

Chapter 3. Tutorials

Not available.

DRAFT

Chapter 4. Architecture and design overview

For a review of the DRS architecture and design please see [Wide Area Data Replication for Scientific Collaboration](#)¹.

DRAFT

¹ <http://www.isi.edu/~annc/papers/grid2005submitted.pdf>

Chapter 5. APIs

1. Programming Model Overview

The DRS is a WS-RF compliant service implemented using the Globus Java WS Core. It exposes a set of Resource Properties and operations to allow users to create replication resources, control replication resources' lifecycle, and inspect the state of replication resources' activities along with the success or failure of individual replicated data sets. In this release, the WSDL and the command-line clients are the primary public interfaces for developers. Two java interfaces exist on the service-side to allow developers and users to modify the source selection behavior of the DRS. These interfaces allow users to chose alternate schemes to select sources beyond the random selection provided by default.

2. Component API

Interfaces to influence source selection include:

- ReplicaCatalogFilter
- SourceSelector

Please see [service-side interfaces](#)¹ for documentation on these interfaces.

¹ <http://www.isi.edu/~schuler/drsdocs401/>

Chapter 6. Services and WSDL

1. Protocol overview

The DRS provides a set of Resource Properties and SOAP operations to create, manipulate and inspect replication activities. Users will begin by creating a replication resource (AKA "Replicator") by invoking the create operation and passing it a URL of the replication request file (described in the domain-specific interface section). Users may start, stop, suspend and resume the Replicator when necessary. Typically a user is expected to simply start the resource and allow it to run through completion. During and after the course of replication activities performed by the resource, users may invoke standard "get resource property" and DRS-specific "find" operations to inspect the state of the resource. When the resource finishes the replication activities and the user has satisfactorily inspected the resource state, the resource should be destroyed using the standard "destroy" operation.

2. Operations

Supported operations include:

- `createReplicator` creates the "Replicator" resource.
 - `[in] InitialTerminationTime` The requested initial termination time for the resource.
 - `[in] requestFileRequest` The request-file style request.
 - `credentialEPR` Endpoint Reference of the user's delegated credential.
 - `options` Replication options which include a set of options pertinent to the transfer stage of the request, such as concurrency, parallel streams, tcp buffer size, etc.
 - `autostart` A Boolean flag indicating whether the resource should be automatically started following resource creation.
 - `requestFileUri` The URI of the request file. Currently supported schemes include http, file, and ftp.
 - `format` The request file format (domain-specific). Currently, the service only supports a simple "Table" format.
 - `[out] EPR` The Endpoint Reference of the Replicator resource.
 - `[fault] fault` Indicates a general failure when attempting to create the Replicator resource.
- `start` starts the resource.
 - `[fault] invalidStateFault` Indicates the resource is in an invalid state to perform the operation.
- `stop` stops the resource.
 - `[fault] invalidStateFault` Indicates the resource is in an invalid state to perform the operation.
- `suspend` suspends the resource.
 - `[fault] invalidStateFault` Indicates the resource is in an invalid state to perform the operation.
- `resume` resumes the resource.

- `[fault] invalidStateFault` Indicates the resource is in an invalid state to perform the operation.
- `findItems` Finds state information for individual replication items.
 - `[in] byUri` Finds by replication URI (currently, this value must be the logical filename, LFN, rather than a properly formed URI). This param is mutually-exclusive with `byStatus`.
 - `[in] byStatus` Find by status, which includes `Pending`, `Finished`, `Failed`, and `Terminated`. This param is mutually-exclusive with `byUri`.
 - `[in] offset` An offset into the results set.
 - `[in] limit` A limit of results to be returned to the client.
 - `[out] items` An array of items to be returned to the client as a result of the find operation. Each item in the array contains the complete status of the replication item including its identifier, priority, status, error (if any), sources, and destinations.
- `[fault] internalErrorFault` Indicates that an internal error occurred.

3. Data Replication Service (DRS) Resource properties

Supported resource properties for DataRep include:

- `status`: The status of the resource, such as `Pending`, `Active`, `Suspended`, `Terminated`, `Destroyed`, etc.
- `stage`: The current stage or activity of the resource, such as `Discover`, `Transfer`, and `Register`.
- `result`: The final result (if any) of the resource, such as `Finished`, `Failed`, and `Exception`.
- `errorMessage`: A verbose description of an error (if any) encountered by the resource. The message may include error or exception information returned by one of the dependent components, such as RLS or RFT.
- `count`: An element containing counts of individual replication items pertaining to total, finished, failed, and terminated replication items.

4. Faults

Supported faults include:

- `CreateReplicatorFault` Indicates that the service failed to create the Replicator resource.
- `RequestBodyMissingFault` Indicates that the request body of the create message parameters was missing.
- `CredentialEprMissingFault` Indicates that the delegated credential EPR was missing from the create message.
- `InvalidStateFault` Indicates that the requested lifecycle operation (e.g., start, stop, suspend, resume) was performed on a resource that was not in the proper state for the operation to succeed (e.g., performing a resume operation on a non-suspended Replicator resource).
- `InternalErrorFaultType` Indicates that an internal error occurred (e.g., internal system failure, etc.).

5. WSDL and Schema Definition

For more information, please see the [Replicator Port Type](#)¹ or the complete list of [schemas](#)².

DRAFT

¹ http://viewcvs.globus.org/viewcvs.cgi/ws-replica/replicator/common/schema/replica/replicator/replicator_port_type.wsdl?rev=1.2.2.1&only_with_tag=globus_4_0_branch&content-type=text/vnd.viewcvs-markup

² http://viewcvs.globus.org/viewcvs.cgi/ws-replica/replicator/common/schema/replica/replicator/?only_with_tag=globus_4_0_branch

DataRep Commands

The DRS provides a set of command-line tools to control the creation and lifecycle of a given replication request. These command line tools are available on Unix and Windows platforms and will work in the same way (of course within the platform rules - the path syntax, variable definitions, etc.).

DRAFT

Name

`globus-replication-create` -- This tool is used to create a replication resource by submitting a replication request to the designated replication service.

```
globus-replication-create
```

Tool description

Use this tool to create replication resources (also referred to as "Replicator" resources). You must specify the URL of the ReplicationService where the resource will be created. You must submit the filename of a file containing an Endpoint Reference (EPR) to a delegated credential resource, which you must have previously created. Finally, you must submit the URL of a request file specifying the desired data replications. If the client is running local to the service container the URL may be a `file://` URL, whereas if the client is remote the URL may be a `http://` or `ftp://` URL. The request file adopts a table format structure where each line in the file represents a source-destination pair delimited by a single *tab* character. The source should be a logical filename (LFN) as found in a Replica Location Service (RLS) Replica Location Index (RLI) service. The destination should be a URL acceptable to the GridFTP server. Most likely, you will want to specify a filename in order to save the newly created Replicator resource's EPR. You may use the EPR for starting the resource and querying its resource properties.

Command syntax

```
globus-replication-create [options] request-file
```

Table 1. Options

-a,--anonymous	Use anonymous authentication. (requires either -m 'conv' or transport (https) security)
--binary <boolean>	Specifies binary data transfer
--blockSize <int>	Block size for data transfer
-c,--serverCertificate <file>	A file with server's certificate used for encryption. Used in the case of GSI Secure Message encryption
-C,--delegatedCredential <file>	Loads Delegated Credential EPR from file
--concurrency <int>	Concurrency of data transfer
-d,--debug	Enables debug mode
--dataChannelAuth <boolean>	Data channel authentication for transfers
--destinationSubject <name>	Destination subject name for data transfer
-e,--eprFile <file>	Loads EPR from file
-f,--descriptor <file>	Sets client security descriptor. Overrides all other security settings
-g,--delegation <mode>	Performs delegation. Can be 'limited' or 'full'. (requires -m 'conv')
-h,--help	Displays help
-k,--key <name value>	Resource Key
-l,--contextLifetime <value>	Lifetime of context created for GSI Secure Conversation (requires -m 'conv')
-m,--securityMech <type>	Sets authentication mechanism: 'msg' (for GSI Secure Message), or 'conv' (for GSI Secure Conversation)
-p,--protection <type>	Sets protection level, can be 'sig' (for signature) can be 'enc' (for encryption)
--parallelStreams <int>	Parallel streams for data transfer
-s,--service <url>	Service URL
-S,--start	Starts the Replicator resource immediately
--sourceSubject <name>	Source subject name for data transfer
--subject <name>	Subject name for data transfer
--tcpBufferSize <int>	TCP buffer size for data transfer
--userName <name>	User name for data transfer
-V,--saveEpr <file>	Save EPR of newly created Replicator to file
-z,--authorization <type>	Sets authorization, can be 'self', 'host' or 'none'

Name

globus-replication-start -- This tool starts the replication activities.

globus-replication-start

Tool description

Replication resources created with the `globus-replication-create` tool may be "started" by using this tool and passing the filename of the saved EPR as a parameter to the tool. The tool will indicate an error condition if the user attempts to start a resource that has been previously started.

Command syntax

```
globus-replication-start [options]
```

Table 2. Options

-a,--anonymous	Use anonymous authentication. (requires either -m 'conv' or transport (https) security)
-c,--serverCertificate <file>	A file with server's certificate used for encryption. Used in the case of GSI Secure Message encryption
-d,--debug	Enables debug mode
-e,--eprFile <file>	Loads EPR from file
-f,--descriptor <file>	Sets client security descriptor. Overrides all other security settings
-g,--delegation <mode>	Performs delegation. Can be 'limited' or 'full'. (requires -m 'conv')
-h,--help	Displays help
-k,--key <name value>	Resource Key
-l,--contextLifetime <value>	Lifetime of context created for GSI Secure Conversation (requires -m 'conv')
-m,--securityMech <type>	Sets authentication mechanism: 'msg' (for GSI Secure Message), or 'conv' (for GSI Secure Conversation)
-p,--protection <type>	Sets protection level, can be 'sig' (for signature) can be 'enc' (for encryption)
-s,--service <url>	Service URL
-z,--authorization <type>	Sets authorization, can be 'self', 'host' or 'none'

Name

globus-replication-stop -- This tool stops the replication activities.

globus-replication-stop

Tool description

Replication resources created with the `globus-replication-create` tool may be "stoped" by using this tool and passing the filename of the saved EPR as a parameter to the tool. The tool will indicate an error condition if the user attempts to stop a resource that has not been previously started, a resource that has been suspended, or a resource that has terminated or been destroyed.

Command syntax

```
globus-replication-stop [options]
```

Table 3. Options

-a,--anonymous	Use anonymous authentication. (requires either -m 'conv' or transport (https) security)
-c,--serverCertificate <file>	A file with server's certificate used for encryption. Used in the case of GSI Secure Message encryption
-d,--debug	Enables debug mode
-e,--eprFile <file>	Loads EPR from file
-f,--descriptor <file>	Sets client security descriptor. Overrides all other security settings
-g,--delegation <mode>	Performs delegation. Can be 'limited' or 'full'. (requires -m 'conv')
-h,--help	Displays help
-k,--key <name value>	Resource Key
-l,--contextLifetime <value>	Lifetime of context created for GSI Secure Conversation (requires -m 'conv')
-m,--securityMech <type>	Sets authentication mechanism: 'msg' (for GSI Secure Message), or 'conv' (for GSI Secure Conversation)
-p,--protection <type>	Sets protection level, can be 'sig' (for signature) can be 'enc' (for encryption)
-s,--service <url>	Service URL
-z,--authorization <type>	Sets authorization, can be 'self', 'host' or 'none'

Name

globus-replication-suspend -- This tool suspends the replication activities.

globus-replication-suspend

Tool description

Replication resources created with the `globus-replication-create` tool may be "suspended" by using this tool and passing the filename of the saved EPR as a parameter to the tool. The tool will indicate an error condition if the user attempts to suspend a resource that has not been previously started, a resource that has been suspended, or a resources that is done or has been destroyed.

Command syntax

```
globus-replication-suspend [options]
```

Table 4. Options

-a,--anonymous	Use anonymous authentication. (requires either -m 'conv' or transport (https) security)
-c,--serverCertificate <file>	A file with server's certificate used for encryption. Used in the case of GSI Secure Message encryption
-d,--debug	Enables debug mode
-e,--eprFile <file>	Loads EPR from file
-f,--descriptor <file>	Sets client security descriptor. Overrides all other security settings
-g,--delegation <mode>	Performs delegation. Can be 'limited' or 'full'. (requires -m 'conv')
-h,--help	Displays help
-k,--key <name value>	Resource Key
-l,--contextLifetime <value>	Lifetime of context created for GSI Secure Conversation (requires -m 'conv')
-m,--securityMech <type>	Sets authentication mechanism: 'msg' (for GSI Secure Message), or 'conv' (for GSI Secure Conversation)
-p,--protection <type>	Sets protection level, can be 'sig' (for signature) can be 'enc' (for encryption)
-s,--service <url>	Service URL
-z,--authorization <type>	Sets authorization, can be 'self', 'host' or 'none'

Name

globus-replication-resume -- This tool resumes the replication activities.

globus-replication-resume

Tool description

Replication resources created with the `globus-replication-create` tool may be "resumed" by using this tool and passing the filename of the saved EPR as a parameter to the tool. The tool will indicate an error condition if the user attempts to resume a resource that has not been previously suspended, or a resource that is done or has been destroyed.

Command syntax

```
globus-replication-resume [options]
```

Table 5. Options

-a,--anonymous	Use anonymous authentication. (requires either -m 'conv' or transport (https) security)
-c,--serverCertificate <file>	A file with server's certificate used for encryption. Used in the case of GSI Secure Message encryption
-d,--debug	Enables debug mode
-e,--eprFile <file>	Loads EPR from file
-f,--descriptor <file>	Sets client security descriptor. Overrides all other security settings
-g,--delegation <mode>	Performs delegation. Can be 'limited' or 'full'. (requires -m 'conv')
-h,--help	Displays help
-k,--key <name value>	Resource Key
-l,--contextLifetime <value>	Lifetime of context created for GSI Secure Conversation (requires -m 'conv')
-m,--securityMech <type>	Sets authentication mechanism: 'msg' (for GSI Secure Message), or 'conv' (for GSI Secure Conversation)
-p,--protection <type>	Sets protection level, can be 'sig' (for signature) can be 'enc' (for encryption)
-s,--service <url>	Service URL
-z,--authorization <type>	Sets authorization, can be 'self', 'host' or 'none'

Name

globus-replication-finditems -- This tool queries the replication resource to return the status of individual replication item activities.

globus-replication-finditems

Tool description

This tool provides the ability to query the status of individual replication items (e.g., replication of a specific file or files) managed by the given Replication resources. It is possible to query for the status of a specific named item or to query for the status of multiple items based on a particular status (e.g., Pending, Finished, Failed). In addition, to reduce potentially large overhead of returning a large results set to the client, the client may specify an offset and limit for the results set to be returned. The "name" or "status" option must be specified.

Command syntax

```
globus-replication-finditems [options] {-N name | -S status}
```

Table 6. Options

-a,--anonymous	Use anonymous authentication. (requires either -m 'conv' or transport (https) security)
-c,--serverCertificate <file>	A file with server's certificate used for encryption. Used in the case of GSI Secure Message encryption
-d,--debug	Enables debug mode
-e,--eprFile <file>	Loads EPR from file
-f,--descriptor <file>	Sets client security descriptor. Overrides all other security settings
-g,--delegation <mode>	Performs delegation. Can be 'limited' or 'full'. (requires -m 'conv')
-h,--help	Displays help
-k,--key <name value>	Resource Key
-l,--contextLifetime <value>	Lifetime of context created for GSI Secure Conversation (requires -m 'conv')
-L,--limit <num>	Limit on the size of the result set.
-m,--securityMech <type>	Sets authentication mechanism: 'msg' (for GSI Secure Message), or 'conv' (for GSI Secure Conversation)
-N,--byName <name>	Finds item by the Logical Filename (LFN) name.
-O,--offset <num>	Offset into the results set. Indexed by 0.
-p,--protection <type>	Sets protection level, can be 'sig' (for signature) can be 'enc' (for encryption)
-S,--byStatus <status>	Finds item(s) by status. Valid status values include "Pending", "Finished", "Failed", and "Terminated".
-s,--service <url>	Service URL
-z,--authorization <type>	Sets authorization, can be 'self', 'host' or 'none'

Chapter 7. Replication request file

1. Interface introduction

The DRS domain-specific interface defines the structure and expected contents of a request file used when creating a replication resource. When the client invokes the create operation of the DRS, it will be expected to submit a properly formatted request file. It is important to understand the structure of the request file and to ensure that the file is well-formed.

2. Syntax of the interface

For the present release, the DRS request file format is fairly trivial. The request file is structured as a "Table" style of rows and columns of text. Each row represent a requested replication item described in two columns. The first column contains the identifier of the data set which will be discovered and replicated. The identifier must be resolvable by the Replica Location Index (see the JNDI configuration for `defaultIndexUrl`). In most cases, it is expected that the identifier be a Logical Filename (LFN) per the Replica Location Service definition. The second column of the row contains the URL of the "destination" for the replication item. The two columns must be delimited by a TAB character and each row must be delimited by an EOL character.



Note

The service will not accept SPACE characters as a substitute for the TAB character.

The following example shows the output of a small request file.

```
% cat example.req
my-lfn-1      gsiftp://myhost:9001/sandbox/examples/files/my-pfn-1
my-lfn-2      gsiftp://myhost:9001/sandbox/examples/files/my-pfn-2
my-lfn-3      gsiftp://myhost:9001/sandbox/examples/files/my-pfn-3
my-lfn-4      gsiftp://myhost:9001/sandbox/examples/files/my-pfn-4
my-lfn-5      gsiftp://myhost:9001/sandbox/examples/files/my-pfn-5
```

Chapter 8. Configuring

This information is in addition to the basic configuration instructions in the [Installing GT 4.2.0](#). Aside from the basic configuration of GT 4.2.0, please review the following instructions:

1. Configuration overview

The DRS requires certain JNDI settings to be properly configured. The installed JNDI configuration file may be found at `$GLOBUS_LOCATION/etc/globus_wsrp_replicator/jndi-config.xml`. To view the default configuration file (shipped with the GT 4.2.0 release) from the Globus CVS repository [click here](#)¹.

2. Syntax of the interface

The settings are structured as name-value pairs. For example:

```
<parameter>
  <name>defaultIndexUrl</name>
  <value>rls://127.0.0.1:39281</value>
</parameter>
```

The following settings must be properly configured:

- `proxyfileDir`: the directory that you would like the DRS to temporarily store user proxies. No setting is necessary. This value may be empty.
- `requestfileDir`: the directory that you would like the DRS to temporarily store request files. No setting is necessary. This value may be empty.
- `defaultIndexUrl`: the connection URL for your installation of RLS running as a RLI service.
- `defaultRegistrationUrl`: the connection URL for your installation of RLS running as a LRC service.
- `defaultReliableTransferUrl`: the connection URL for your installation of the RFT ReliableFileTransferFactoryService.
- `proxyfileChangePermsCmd`: the platform-dependent command to change file permissions to user-only read-write permissions.
- The rest of the parameter/value pairs may retain the given default values.

¹ http://viewcvs.globus.org/viewcvs.cgi/ws-replica/replicator/service/java/source/deploy-jndi-config.xml?tab=globus_4_2_branch&content-type=text/vnd.viewcvs-markup

Chapter 9. Environment variable interface

1. Environmental variables for DRS

- `GLOBUS_LOCATION=/path/to/globus/install`

DRAFT

Chapter 10. Debugging

Log output from DRS is a useful tool for debugging issues. Because DRS is built on top of Java WS Core, developer debugging is the same as described in [Chapter 10, Debugging](#). You can also find information about sys admin logging in [Chapter 5, Debugging](#).

1. Development Logging in Java WS Core

The following information applies to Java WS Core and those services built on it.

Logging in the Java WS Core is based on the [Jakarta Commons Logging](#)¹ API. Commons Logging provides a consistent interface for instrumenting source code while at the same time allowing the user to plug-in a different logging implementation. Currently we use [Log4j](#)² as a logging implementation. Log4j uses a separate configuration file to configure itself. Please see Log4j documentation for details on the [configuration file format](#)³.

1.1. Configuring server side developer logs

Server side logging can be configured in `$GLOBUS_LOCATION/container-log4j.properties`, when the container is stand alone container. For tomcat level logging, refer to [Logging for Tomcat](#)⁴, . The logger `log4j.appender.A1` is used for developer logging and by default writes output to the system output. By default it is set for all warnings in the Globus Toolkit package to be displayed.

Additional logging can be enabled for a package by adding a new line to the configuration file. Example:

```
#for debug level logging from org.globus.package.FooClass
log4j.category.org.globus.package.name.FooClass=DEBUG
#for warnings from org.some.warn.package
log4j.category.org.some.warn.package=WARN
```

1.2. Configuring client side developer logs

Client side logging can be configured in `$GLOBUS_LOCATION/log4j.properties`. The logger `log4j.appender.A1` is used for developer logging and by default writes output to the system output. By default it is set for all warnings in the Globus Toolkit package to be displayed.

2. Enable Debug Logging for DRS

To turn on debug logging for DRS, add the line:

```
log4j.category.org.globus.replica=DEBUG
```

to the appropriate properties file.

¹ <http://jakarta.apache.org/commons/logging/>

² <http://logging.apache.org/log4j/>

³ [http://logging.apache.org/log4j/docs/api/org/apache/log4j/PropertyConfigurator.html#doConfigure\(java.lang.String,org.apache.log4j.spi.LoggerRepository\)](http://logging.apache.org/log4j/docs/api/org/apache/log4j/PropertyConfigurator.html#doConfigure(java.lang.String,org.apache.log4j.spi.LoggerRepository))

⁴ <http://tomcat.apache.org/tomcat-5.5-doc/logging.html>

Chapter 11. Troubleshooting

For a list of common errors in GT, see [Error Codes](#).

1. Errors

Table 11.1. Data Replication Service (DRS) Errors

Error Code	Definition	Possible Solutions
Authorization failed. Expected <hostname1> target but received <hostname2>	Did not receive expected hostname	When authorization is enabled on the container, you may need to use the proper hostname when referencing the DRS service rather than using localhost.
org.globus.wsrp.ResourceException: Failed to create Replication: /scratch/testrun (No such file or directory)	Cannot find the request file	Ensure that the request file's filename is correct, that it is reachable by the DRS service, and that it has the appropriate permissions for the DRS service to access it.
org.globus.wsrp.ResourceException: Failed to create Replication: String index out of range: -1	The request file is malformed (for example by using spaces instead of a delimiting tab character) which is resulting in a runtime exception.	Make sure your request file is in the correct form as described here .

Chapter 12. Related Documentation

Not available.

DRAFT

Index

A

architecture, 5

D

debugging

 logging, 21

E

errors, 22

L

logging, 21

 debugging, 21

R

replication activities

 resuming, 16

 stopping, 14

 suspending, 15

replication resource

 creating, 11

 finding replication item status, 17

 starting, 13

DRAFT

GT 4.2.0 Migrating Guide for Data Replication Service (DRS)

Table of Contents

1. Migrating from GT4.0	1
2. Migrating from GT3	1
3. Migrating from GT2	1

<titleabbrev>Migrating Guide</titleabbrev>

The following provides available information about migrating from previous versions of the Globus Toolkit.

1. Migrating from GT4.0

The public interfaces to Java WS Core (on which DRS is based) have changed and are not backwards compatible. For information on updating your services, see [Section 1, “Updating higher level services”](#).

2. Migrating from GT3

This component is new with GT version 4.0.0; therefore, this section does not apply.

3. Migrating from GT2

This component is new with GT version 4.0.0; therefore, this section does not apply.

GT 4.2.0 Data Replication Service: Quality Profile

Table of Contents

1. Test coverage reports	1
2. Code analysis reports	1
3. Outstanding bugs	1
4. Bug Fixes	1
5. Performance reports	1

<titleabbrev>Quality Profile</titleabbrev>

1. Test coverage reports

Not available.

2. Code analysis reports

Not available.

3. Outstanding bugs

- [Bug 3502](#)¹: Container freezes (CPU spin) when making remote call between ws-resources. See record for *WORKAROUND*.
- [Bug 4231](#)²: DRS does not implement subscription/notification.

See a [bugzilla](#)³ query on Replication Services, DRS to list the bugs outstanding.

4. Bug Fixes

- [Bug 5495](#)⁴: Java spec upgrade
- [Bug 5810](#)⁵: null exception when Saml Callout authorization is configured

5. Performance reports

For a review of the DRS performance, please see [Wide Area Data Replication for Scientific Collaboration](#)⁶.

¹ http://bugzilla.globus.org/globus/show_bug.cgi?id=3502

² http://bugzilla.globus.org/globus/show_bug.cgi?id=4231

³ <http://bugzilla.globus.org/globus/query.cgi>

⁴ http://bugzilla.globus.org/bugzilla/show_bug.cgi?id=5495

⁵ http://bugzilla.globus.org/bugzilla/show_bug.cgi?id=5810

⁶ <http://www.isi.edu/~annc/papers/grid2005submitted.pdf>

GT 4.2.0 Release Notes: Data Replication Service (DRS)

Table of Contents

1. Component Overview	1
2. Feature summary	1
3. Summary of Changes in DRS	2
4. Bug Fixes	2
5. Known Problems	2
6. Technology dependencies	2
7. Tested platforms	3
8. Backward compatibility summary	3
9. Associated Standards	3
10. For More Information	3

<titleabbrev>Release Notes</titleabbrev>

1. Component Overview

The Data Replication Service (DRS) is provided with the Globus Toolkit 4.2.0 and first appeared in the GT 3.9.5 Beta release. The primary functionality of the component allows users to identify a set of desired files existing in their Grid environment, to make local replicas of those data files by transferring files from one or more source locations, and to register the new replicas in a Replica Location Service. The DRS conforms to the WS-RF specification and exposes a WS-Resource (called a "Replicator" resource) which represents the current state of the requested replication activity and allows users to query or subscribe to various Resource Properties in order to monitor the state of the resource. The DRS is built on the GT 4.2.0 Java WS Core and uses the Globus RLS to locate and register replicas and the Globus RFT to transfer files.

2. Feature summary

Features new in release GT 4.2.0:

- None.

Other Supported Features

- Improved implementation of the Data Replication Service: a WS-Resource, called the *Replicator*, which accepts a request from a client to locate, transfer, and register new replicas of data files in the Grid environment.
- A set of command-line tools to create (`globus-replication-create`), start (`globus-replication-start`), stop (`globus-replication-stop`), suspend (`globus-replication-suspend`), resume (`globus-replication-resume`) replication requests, and find item status (`globus-replication-finditems`).
- WSDL-defined SOAP operations to *create*, *start*, *stop*, *suspend*, and *resume* a replication request, along with operations to get the status of individual replicas in the request. For details, [click here](#)¹ to view a listing of the WSDL-defined interface from the Globus CVS repository.

¹ http://viewcvs.globus.org/viewcvs.cgi/ws-replica/replicator/common/schema/replica/replicator/?only_with_tag=globus_4_0_branch

- APIs to allow users to implement custom replica source selection algorithms.
- Supports secure transport, secure conversation, and secure message communication as provided by GT 4.2.0.

Deprecated Features

- Database-backed State Persistence: State is now maintained in memory and lasts only for the lifetime of the WS-Resource or as dictated by the service container. This change simplifies setup of the DRS. We intend to reintroduce other persistence model(s) after we have collected additional user feedback on the DRS.

3. Summary of Changes in DRS

The Data Replication Service (DRS) has been updated to conform to the Java spec upgrade (bug 5495) and Security restructuring (bug 5559) for Java WS Core.

4. Bug Fixes

- [Bug 5495](#)²: Java spec upgrade
- [Bug 5810](#)³: null exception when SAML Callout authorization is configured

5. Known Problems

The following problems and limitations are known to exist for the Data Replication Service at the time of the 4.2.0 release:

5.1. Limitations

- No known limitations exist.

5.2. Outstanding bugs

- [Bug 3502](#)⁴: Container freezes (CPU spin) when making remote call between ws-resources. See record for *WORKAROUND*.
- [Bug 4231](#)⁵: DRS does not implement subscription/notification.

See a [bugzilla](#)⁶ query on Replication Services, DRS to list the bugs outstanding.

6. Technology dependencies

DRS depends on the following GT components:

- Java WS Core
- WS Authentication and Authorization

² http://bugzilla.globus.org/bugzilla/show_bug.cgi?id=5495

³ http://bugzilla.globus.org/bugzilla/show_bug.cgi?id=5810

⁴ http://bugzilla.globus.org/globus/show_bug.cgi?id=3502

⁵ http://bugzilla.globus.org/globus/show_bug.cgi?id=4231

⁶ <http://bugzilla.globus.org/globus/query.cgi>

- Delegation Service
- RFT
- RLS

DRS depends on the following 3rd party software:

- None

7. Tested platforms

Tested Platforms for DRS

- Linux (RedHat, Debian)

8. Backward compatibility summary

Protocol changes since GT version 4.0.x:

- None

API changes since GT version 4.0.x:

- None

Exception changes since GT version 4.0.x:

- None

Schema changes since GT version 4.0.x:

- None

9. Associated Standards

Associated standards for DataRep:

- [WS-RF](#)⁷
- [WS-Addressing](#)⁸
- [WS-Security](#)⁹

10. For More Information

Click [here](#) for more information about this component.

⁷ <http://docs.oasis-open.org/wsrf/2004/06/wsrf-WS-ServiceGroup-1.2-draft-02.pdf>

⁸ <http://msdn.microsoft.com/ws/2004/03/ws-addressing>

⁹ <http://msdn.microsoft.com/webservices/understanding/specs/default.aspx?pull=/library/en-us/dnglobspec/html/wssecurspecindex.asp>