

# 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](#)<sup>1</sup> 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:

---

<sup>1</sup> [http://viewcvs.globus.org/viewcvs.cgi/ws-replica/replicator/common/schema/replica/replicator/?only\\_with\\_tag=globus\\_4\\_0\\_branch](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](#)<sup>1</sup>.

DRAFT

---

<sup>1</sup> <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](#)<sup>1</sup> for documentation on these interfaces.

---

<sup>1</sup> <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](#)<sup>1</sup> or the complete list of [schemas](#)<sup>2</sup>.

DRAFT

---

<sup>1</sup> [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](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)

<sup>2</sup> [http://viewcvs.globus.org/viewcvs.cgi/ws-replica/replicator/common/schema/replica/replicator/?only\\_with\\_tag=globus\\_4\\_0\\_branch](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**

<b>-a,--anonymous</b>	Use anonymous authentication. (requires either -m 'conv' or transport (https) security)
<b>--binary &lt;boolean&gt;</b>	Specifies binary data transfer
<b>--blockSize &lt;int&gt;</b>	Block size for data transfer
<b>-c,--serverCertificate &lt;file&gt;</b>	A file with server's certificate used for encryption. Used in the case of GSI Secure Message encryption
<b>-C,--delegatedCredential &lt;file&gt;</b>	Loads Delegated Credential EPR from file
<b>--concurrency &lt;int&gt;</b>	Concurrency of data transfer
<b>-d,--debug</b>	Enables debug mode
<b>--dataChannelAuth &lt;boolean&gt;</b>	Data channel authentication for transfers
<b>--destinationSubject &lt;name&gt;</b>	Destination subject name for data transfer
<b>-e,--eprFile &lt;file&gt;</b>	Loads EPR from file
<b>-f,--descriptor &lt;file&gt;</b>	Sets client security descriptor. Overrides all other security settings
<b>-g,--delegation &lt;mode&gt;</b>	Performs delegation. Can be 'limited' or 'full'. (requires -m 'conv')
<b>-h,--help</b>	Displays help
<b>-k,--key &lt;name value&gt;</b>	Resource Key
<b>-l,--contextLifetime &lt;value&gt;</b>	Lifetime of context created for GSI Secure Conversation (requires -m 'conv')
<b>-m,--securityMech &lt;type&gt;</b>	Sets authentication mechanism: 'msg' (for GSI Secure Message), or 'conv' (for GSI Secure Conversation)
<b>-p,--protection &lt;type&gt;</b>	Sets protection level, can be 'sig' (for signature) can be 'enc' (for encryption)
<b>--parallelStreams &lt;int&gt;</b>	Parallel streams for data transfer
<b>-s,--service &lt;url&gt;</b>	Service URL
<b>-S,--start</b>	Starts the Replicator resource immediately
<b>--sourceSubject &lt;name&gt;</b>	Source subject name for data transfer
<b>--subject &lt;name&gt;</b>	Subject name for data transfer
<b>--tcpBufferSize &lt;int&gt;</b>	TCP buffer size for data transfer
<b>--userName &lt;name&gt;</b>	User name for data transfer
<b>-V,--saveEpr &lt;file&gt;</b>	Save EPR of newly created Replicator to file
<b>-z,--authorization &lt;type&gt;</b>	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**

<b>-a,--anonymous</b>	Use anonymous authentication. (requires either -m 'conv' or transport (https) security)
<b>-c,--serverCertificate &lt;file&gt;</b>	A file with server's certificate used for encryption. Used in the case of GSI Secure Message encryption
<b>-d,--debug</b>	Enables debug mode
<b>-e,--eprFile &lt;file&gt;</b>	Loads EPR from file
<b>-f,--descriptor &lt;file&gt;</b>	Sets client security descriptor. Overrides all other security settings
<b>-g,--delegation &lt;mode&gt;</b>	Performs delegation. Can be 'limited' or 'full'. (requires -m 'conv')
<b>-h,--help</b>	Displays help
<b>-k,--key &lt;name value&gt;</b>	Resource Key
<b>-l,--contextLifetime &lt;value&gt;</b>	Lifetime of context created for GSI Secure Conversation (requires -m 'conv')
<b>-m,--securityMech &lt;type&gt;</b>	Sets authentication mechanism: 'msg' (for GSI Secure Message), or 'conv' (for GSI Secure Conversation)
<b>-p,--protection &lt;type&gt;</b>	Sets protection level, can be 'sig' (for signature) can be 'enc' (for encryption)
<b>-s,--service &lt;url&gt;</b>	Service URL
<b>-z,--authorization &lt;type&gt;</b>	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**

<b>-a,--anonymous</b>	Use anonymous authentication. (requires either -m 'conv' or transport (https) security)
<b>-c,--serverCertificate &lt;file&gt;</b>	A file with server's certificate used for encryption. Used in the case of GSI Secure Message encryption
<b>-d,--debug</b>	Enables debug mode
<b>-e,--eprFile &lt;file&gt;</b>	Loads EPR from file
<b>-f,--descriptor &lt;file&gt;</b>	Sets client security descriptor. Overrides all other security settings
<b>-g,--delegation &lt;mode&gt;</b>	Performs delegation. Can be 'limited' or 'full'. (requires -m 'conv')
<b>-h,--help</b>	Displays help
<b>-k,--key &lt;name value&gt;</b>	Resource Key
<b>-l,--contextLifetime &lt;value&gt;</b>	Lifetime of context created for GSI Secure Conversation (requires -m 'conv')
<b>-m,--securityMech &lt;type&gt;</b>	Sets authentication mechanism: 'msg' (for GSI Secure Message), or 'conv' (for GSI Secure Conversation)
<b>-p,--protection &lt;type&gt;</b>	Sets protection level, can be 'sig' (for signature) can be 'enc' (for encryption)
<b>-s,--service &lt;url&gt;</b>	Service URL
<b>-z,--authorization &lt;type&gt;</b>	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**

<b>-a,--anonymous</b>	Use anonymous authentication. (requires either -m 'conv' or transport (https) security)
<b>-c,--serverCertificate &lt;file&gt;</b>	A file with server's certificate used for encryption. Used in the case of GSI Secure Message encryption
<b>-d,--debug</b>	Enables debug mode
<b>-e,--eprFile &lt;file&gt;</b>	Loads EPR from file
<b>-f,--descriptor &lt;file&gt;</b>	Sets client security descriptor. Overrides all other security settings
<b>-g,--delegation &lt;mode&gt;</b>	Performs delegation. Can be 'limited' or 'full'. (requires -m 'conv')
<b>-h,--help</b>	Displays help
<b>-k,--key &lt;name value&gt;</b>	Resource Key
<b>-l,--contextLifetime &lt;value&gt;</b>	Lifetime of context created for GSI Secure Conversation (requires -m 'conv')
<b>-m,--securityMech &lt;type&gt;</b>	Sets authentication mechanism: 'msg' (for GSI Secure Message), or 'conv' (for GSI Secure Conversation)
<b>-p,--protection &lt;type&gt;</b>	Sets protection level, can be 'sig' (for signature) can be 'enc' (for encryption)
<b>-s,--service &lt;url&gt;</b>	Service URL
<b>-z,--authorization &lt;type&gt;</b>	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**

<b>-a,--anonymous</b>	Use anonymous authentication. (requires either -m 'conv' or transport (https) security)
<b>-c,--serverCertificate &lt;file&gt;</b>	A file with server's certificate used for encryption. Used in the case of GSI Secure Message encryption
<b>-d,--debug</b>	Enables debug mode
<b>-e,--eprFile &lt;file&gt;</b>	Loads EPR from file
<b>-f,--descriptor &lt;file&gt;</b>	Sets client security descriptor. Overrides all other security settings
<b>-g,--delegation &lt;mode&gt;</b>	Performs delegation. Can be 'limited' or 'full'. (requires -m 'conv')
<b>-h,--help</b>	Displays help
<b>-k,--key &lt;name value&gt;</b>	Resource Key
<b>-l,--contextLifetime &lt;value&gt;</b>	Lifetime of context created for GSI Secure Conversation (requires -m 'conv')
<b>-m,--securityMech &lt;type&gt;</b>	Sets authentication mechanism: 'msg' (for GSI Secure Message), or 'conv' (for GSI Secure Conversation)
<b>-p,--protection &lt;type&gt;</b>	Sets protection level, can be 'sig' (for signature) can be 'enc' (for encryption)
<b>-s,--service &lt;url&gt;</b>	Service URL
<b>-z,--authorization &lt;type&gt;</b>	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**

<b>-a,--anonymous</b>	Use anonymous authentication. (requires either -m 'conv' or transport (https) security)
<b>-c,--serverCertificate &lt;file&gt;</b>	A file with server's certificate used for encryption. Used in the case of GSI Secure Message encryption
<b>-d,--debug</b>	Enables debug mode
<b>-e,--eprFile &lt;file&gt;</b>	Loads EPR from file
<b>-f,--descriptor &lt;file&gt;</b>	Sets client security descriptor. Overrides all other security settings
<b>-g,--delegation &lt;mode&gt;</b>	Performs delegation. Can be 'limited' or 'full'. (requires -m 'conv')
<b>-h,--help</b>	Displays help
<b>-k,--key &lt;name value&gt;</b>	Resource Key
<b>-l,--contextLifetime &lt;value&gt;</b>	Lifetime of context created for GSI Secure Conversation (requires -m 'conv')
<b>-L,--limit &lt;num&gt;</b>	Limit on the size of the result set.
<b>-m,--securityMech &lt;type&gt;</b>	Sets authentication mechanism: 'msg' (for GSI Secure Message), or 'conv' (for GSI Secure Conversation)
<b>-N,--byName &lt;name&gt;</b>	Finds item by the Logical Filename (LFN) name.
<b>-O,--offset &lt;num&gt;</b>	Offset into the results set. Indexed by 0.
<b>-p,--protection &lt;type&gt;</b>	Sets protection level, can be 'sig' (for signature) can be 'enc' (for encryption)
<b>-S,--byStatus &lt;status&gt;</b>	Finds item(s) by status. Valid status values include "Pending", "Finished", "Failed", and "Terminated".
<b>-s,--service &lt;url&gt;</b>	Service URL
<b>-z,--authorization &lt;type&gt;</b>	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](#)<sup>1</sup>.

## 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.

---

<sup>1</sup> [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](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](#)<sup>1</sup> 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](#)<sup>2</sup> as a logging implementation. Log4j uses a separate configuration file to configure itself. Please see Log4j documentation for details on the [configuration file format](#)<sup>3</sup>.

### 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](#)<sup>4</sup>, . 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.

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

<sup>2</sup> <http://logging.apache.org/log4j/>

<sup>3</sup> [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))

<sup>4</sup> <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 <a href="#">here</a> .

---

# 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