

GT 4.2.0 Java WS Core : User's Guide

DRAFT

GT 4.2.0 Java WS Core : User's Guide

Introduction

The Java WS Core User's Guide provides general end user-oriented information.

DRAFT

Table of Contents

1. Using WS Core Commands (Java)	1
1. What is the Java WS Core container?	1
2. Starting the container	1
3. Stopping the container	1
4. GT web services based on Java WS Core	1
5. Querying a resource	1
I. Java WS Core Commands	?
globus-start-container	4
globus-stop-container	5
globus-start-container-detached	7
globus-stop-container-detached	8
wsrf-destroy	9
wsrf-set-termination-time	11
wsrf-query	13
wsrf-get-property	15
wsrf-get-properties	17
wsrf-insert-property	19
wsrf-update-property	21
wsrf-delete-property	23
wsn-get-current-message	25
wsn-pause-subscription	27
wsn-resume-subscription	29
wsn-subscribe	31
globus-deploy-gar	33
globus-undeploy-gar	35
globus-check-environment	37
globus-check-remote-environment	38
globus-update-client-config	39
globus-validate-descriptors	40
globus-reload-container	41
globus-remote-undeploy-gar	43
globus-remote-deploy-gar	45
ws-enumerate-start	47
ws-enumerate	49
ws-enumerate-end	51
globus-xpath-query	53
Common Java Client Options	56
2. Logging	57
3. Troubleshooting	58
1. The standalone container appears to hang	58
2. Java WS Core Errors	59
3. General troubleshooting information	61
A. Running client programs from command line	62
1. With <code>globus-devel-env</code> script help	62
2. Using bootstrap	62
B. Running client programs from any directory	63
C. Usage statistics collection by the Globus Alliance	64
1. Usage statistics sent by Java WS Core	64
Glossary	65
Index	66

List of Tables

1. Options	4
2. Common options	6
3. Shutdown options	6
4. Options	7
5. Options	8
6. Common options	10
7. Command-specific options	11
8. Common options	12
9. Common options	14
10. Common options	16
11. Common options	18
12. Common options	20
13. Common options	22
14. Common options	24
15. Common options	26
16. Common options	28
17. Common options	30
18. Command-specific options	31
19. Common options	32
20. Options	33
21. Supported property-value pairs	33
22. Options	35
23. Options	38
24. Options	39
25. Options	40
26. Common options	42
27. Common options	44
28. Command-specific options	45
29. Common options	46
30. Common options	48
31. Command-specific options	49
32. Common options	50
33. Common options	52
34. Command-specific options	53
35. Common options	54
36. Common options	56
3.1. Java WS Core Errors	60

Chapter 1. Using WS Core Commands (Java)

For a complete list of commands, see [Java WS Core Commands](#).

1. What is the Java WS Core container?

The Java WS Core container is the web services hosting environment based on Java on which the GT higher-level Java web services (such as RFT and CAS) are based.

2. Starting the container

To start the Java WS Core container in any default installation of GT, run `globus-start-container`:

```
$GLOBUS_LOCATION bin/globus-start-container
```

If you want to run without transport-level security, use the `-nosec` option:

```
$GLOBUS_LOCATION bin/globus-start-container -nosec
```

3. Stopping the container

To stop the container, run:

```
$GLOBUS_LOCATION bin/globus-stop-container
```

4. GT web services based on Java WS Core

The following GT components are higher-level web services based on Java WS Core

- [Reliable File Transfer \(RFT\)](#)
- [WS Replica Locator Service \(WS RLS\)](#)
- [GRAM4](#)
- [WS Java Security](#)
- [Community Authorization Service \(CAS\)](#)
- [Monitoring and Discovery System \(WS MDS\)](#)
- [Delegation Service](#)

5. Querying a resource

You can use the `wsrf-query` command to query any WSRF resource property document. For example, you can use the following command to query the WS MDS Index Service for all the resource properties collected by the default Index Service on your local host:

```
$GLOBUS_LOCATION/bin/wsrf-query -s http://localhost:8443/wsrf/services/DefaultIndexService
```

DRAFT

Java WS Core Commands

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.).

The wsrf-* and wsn-* clients should work against any service that supports the given WSRF or WSN operations.

DRAFT

Name

globus-start-container -- Starts standalone container

globus-start-container

Tool description

Starts a standalone container. By default a secure container is started on port 8443 and is accessible via HTTPS. On successful startup a list of services will be displayed on the console. By default the non secure (HTTP) container is started on port 8080.

Command syntax

```
globus-start-container [options]
```

Table 1. Options

-help	Displays help information about the command.
-p <port>	Sets the port number for the container.
-i <address>	Binds container to the specified network address.
-quiet	Does not show a list of services at startup.
-debug	Enables debug mode.
-nosec	Starts a non secure (HTTP) container. Please note that this option only disables transport security. Message security can still be used.
-containerDesc <file>	Specifies a container security descriptor file.
-profile <name>	Specifies a configuration profile name for the container.

Name

`globus-stop-container --` Stops standalone container

`globus-stop-container`

Tool description

Stops a standalone container. By default this command will attempt to stop a container running on **localhost:8443** and perform a **soft** shutdown.

The **globus-stop-container** command invokes a **ShutdownService** running in the container. By default that service is configured to perform **self** authorization and therefore the **globus-stop-container** must be executed with the same credentials as the container it is running with. Alternatively, the service can be configured with a gridmap file to allow a subset of users (with their own credentials) to invoke the service (please see the service security deployment descriptor for details).

Command syntax

`globus-stop-container [options] ['soft' | 'hard']`

Table 2. Common options

-h, --help	Displays help information about the command.
-d, --debug	Enables debug mode. For example, full stack traces of errors will be displayed.
-e, --eprFile <file>	Specifies an <i>XML</i> file that contains the <i>WS-Addressing</i> endpoint reference.
-s, --service <url>	Specifies the service URL.
-k, --key <name value>	Specifies the resource key. The name is the QName of the resource key in the string form: {namespaceURI}localPart , while the value is the simple value of the key. For complex keys, use the --eprFile option. Example: <pre>-k "{http://www.globus.org}MyKey" 123</pre>
-f, --descriptor <file>	Specifies a client security descriptor. Overrides all other security settings.
-a, --anonymous	Enables anonymous authentication. Only supported with transport security or the GSI Secure Conversation authentication mechanism.
-g, --delegation <mode>	Enables delegation. mode can be either 'limited' or 'full' . Only supported with the GSI Secure Conversation authentication mechanism.
-l, --contextLifetime <value>	Sets the lifetime of the client security context. value is in milliseconds. Only supported with the GSI Secure Conversation authentication mechanism.
-m, --securityMech <type>	Specifies the authentication mechanism. type can be 'msg' for GSI Secure Message, or 'conv' for GSI Secure Conversation.
-c, --serverCertificate <file>	Specifies the server's <i>certificate</i> file used for encryption. Only needed for the GSI Secure Message authentication mechanism.
-p, --protection <type>	Specifies the protection level. type can be 'sig' for signature or 'enc' for encryption.
-x, --proxyFilename <value>	Sets the proxy file to use as client credential.
-z, --authorization <type>	Specifies authorization type. type can be 'self' , 'host' , 'none' , or a string specifying the expected identity of the remote party.
-t, --timeout <timeout>	Specifies client timeout (in seconds). The client will wait maximum of the timeout value for a response from the server before returning an error. By default the timeout value is 10 minutes.

Table 3. Shutdown options

'soft'	This option lets the threads die naturally.
'hard'	This option forces an immediate JVM shutdown.

Example:

```
$ globus-stop-container soft
```

Please see the [troubleshooting section](#) if you are having problems with `globus-stop-container`.

Name

`globus-start-container-detached --` Starts standalone container detached from controlling TTY

`globus-start-container-detached`

Tool description

Starts a standalone container detached from the controlling TTY. This can be useful for long running containers or when started from `init.d` scripts. Container log goes to `$GLOBUS_LOCATION/var/container.log` and a PID file is written to `$GLOBUS_LOCATION/var/container.pid`. `globus-start-container-detached` is just a wrapper around `globus-start-container` so see [globus-start-container](#) for more information and options.



Note

Note that this tool is only available after doing a full Globus install. It is not available in Java WS Core only installs.

Command syntax

```
globus-start-container-detached [options] | [arguments to container]
```

Table 4. Options

-logfile <file>	Specifies an alternate container log file.
-append	Do not overwrite the existing log file.
-pidfile <file>	Specifies an alternate PID file location.

Name

`globus-stop-container-detached --` Stops standalone container detached from controlling TTY

`globus-stop-container-detached`

Tool description

Stops a standalone container detached from the controlling TTY. `$GLOBUS_LOCATION/var/container.pid` is used to find the PID of the running container and signals are sent to stop the container.



Note

Note that this tool is only available after doing a full Globus install. It is not available in Java WS Core only installs.

Command syntax

`globus-stop-container-detached [options]`

Table 5. Options

<code>-pidfile <file></code>	Specifies an alternate PID file location.
------------------------------------	---

Name

`wsrf-destroy --` Destroys a resource

`wsrf-destroy`

Tool description

Destroys a resource.

Command syntax

`wsrf-destroy [options]`

DRAFT

Table 6. Common options

-h, --help	Displays help information about the command.
-d, --debug	Enables debug mode. For example, full stack traces of errors will be displayed.
-e, --eprFile <file>	Specifies an <i>XML</i> file that contains the <i>WS-Addressing</i> endpoint reference.
-s, --service <url>	Specifies the service URL.
-k, --key <name value>	Specifies the resource key. The name is the QName of the resource key in the string form: {namespaceURI}localPart , while the value is the simple value of the key. For complex keys, use the --eprFile option. Example: <pre>-k "{http://www.globus.org}MyKey" 123</pre>
-f, --descriptor <file>	Specifies a client security descriptor. Overrides all other security settings.
-a, --anonymous	Enables anonymous authentication. Only supported with transport security or the GSI Secure Conversation authentication mechanism.
-g, --delegation <mode>	Enables delegation. mode can be either 'limited' or 'full' . Only supported with the GSI Secure Conversation authentication mechanism.
-l, --contextLifetime <value>	Sets the lifetime of the client security context. value is in milliseconds. Only supported with the GSI Secure Conversation authentication mechanism.
-m, --securityMech <type>	Specifies the authentication mechanism. type can be 'msg' for GSI Secure Message, or 'conv' for GSI Secure Conversation.
-c, --serverCertificate <file>	Specifies the server's <i>certificate</i> file used for encryption. Only needed for the GSI Secure Message authentication mechanism.
-p, --protection <type>	Specifies the protection level. type can be 'sig' for signature or 'enc' for encryption.
-x, --proxyFilename <value>	Sets the proxy file to use as client credential.
-z, --authorization <type>	Specifies authorization type. type can be 'self' , 'host' , 'none' , or a string specifying the expected identity of the remote party.
-t, --timeout <timeout>	Specifies client timeout (in seconds). The client will wait maximum of the timeout value for a response from the server before returning an error. By default the timeout value is 10 minutes.

Example:

```
$ wsrfl-destroy -s http://localhost:8080/wsrfl/services/CounterService \ -k
  "{http://counter.com}CounterKey" 123
```

Name

`wsrf-set-termination-time` -- Sets termination time of a resource

`wsrf-set-termination-time`

Tool description

Sets a termination time of a resource.

Command syntax

`wsrf-set-termination-time` [options] <seconds> | 'infinity'

The following are command-specific options in addition to the common options:

Table 7. Command-specific options

-u, --utc	Display time in UTC.
------------------	----------------------

Table 8. Common options

-h, --help	Displays help information about the command.
-d, --debug	Enables debug mode. For example, full stack traces of errors will be displayed.
-e, --eprFile <file>	Specifies an <i>XML</i> file that contains the <i>WS-Addressing</i> endpoint reference.
-s, --service <url>	Specifies the service URL.
-k, --key <name value>	Specifies the resource key. The name is the QName of the resource key in the string form: {namespaceURI}localPart , while the value is the simple value of the key. For complex keys, use the --eprFile option. Example: <pre>-k "{http://www.globus.org}MyKey" 123</pre>
-f, --descriptor <file>	Specifies a client security descriptor. Overrides all other security settings.
-a, --anonymous	Enables anonymous authentication. Only supported with transport security or the GSI Secure Conversation authentication mechanism.
-g, --delegation <mode>	Enables delegation. mode can be either 'limited' or 'full' . Only supported with the GSI Secure Conversation authentication mechanism.
-l, --contextLifetime <value>	Sets the lifetime of the client security context. value is in milliseconds. Only supported with the GSI Secure Conversation authentication mechanism.
-m, --securityMech <type>	Specifies the authentication mechanism. type can be 'msg' for GSI Secure Message, or 'conv' for GSI Secure Conversation.
-c, --serverCertificate <file>	Specifies the server's <i>certificate</i> file used for encryption. Only needed for the GSI Secure Message authentication mechanism.
-p, --protection <type>	Specifies the protection level. type can be 'sig' for signature or 'enc' for encryption.
-x, --proxyFilename <value>	Sets the proxy file to use as client credential.
-z, --authorization <type>	Specifies authorization type. type can be 'self' , 'host' , 'none' , or a string specifying the expected identity of the remote party.
-t, --timeout <timeout>	Specifies client timeout (in seconds). The client will wait maximum of the timeout value for a response from the server before returning an error. By default the timeout value is 10 minutes.

Example:

```
$ wsrfl-set-termination-time -s http://localhost:8080/wsrfl/services/CounterService \ -k
  "{http://counter.com}CounterKey" 123 30
```

Name

`wsrf-query` -- Performs query on a resource property document

`wsrf-query`

Tool description

Queries the resource property document of a resource. By default, a simple XPath query is assumed that returns the entire resource property document.

Command syntax

`wsrf-query` [options] [query expression] [dialect]

DRAFT

Table 9. Common options

-h, --help	Displays help information about the command.
-d, --debug	Enables debug mode. For example, full stack traces of errors will be displayed.
-e, --eprFile <file>	Specifies an <i>XML</i> file that contains the <i>WS-Addressing</i> endpoint reference.
-s, --service <url>	Specifies the service URL.
-k, --key <name value>	Specifies the resource key. The name is the QName of the resource key in the string form: {namespaceURI}localPart , while the value is the simple value of the key. For complex keys, use the --eprFile option. Example: <pre>-k "{http://www.globus.org}MyKey" 123</pre>
-f, --descriptor <file>	Specifies a client security descriptor. Overrides all other security settings.
-a, --anonymous	Enables anonymous authentication. Only supported with transport security or the GSI Secure Conversation authentication mechanism.
-g, --delegation <mode>	Enables delegation. mode can be either 'limited' or 'full' . Only supported with the GSI Secure Conversation authentication mechanism.
-l, --contextLifetime <value>	Sets the lifetime of the client security context. value is in milliseconds. Only supported with the GSI Secure Conversation authentication mechanism.
-m, --securityMech <type>	Specifies the authentication mechanism. type can be 'msg' for GSI Secure Message, or 'conv' for GSI Secure Conversation.
-c, --serverCertificate <file>	Specifies the server's <i>certificate</i> file used for encryption. Only needed for the GSI Secure Message authentication mechanism.
-p, --protection <type>	Specifies the protection level. type can be 'sig' for signature or 'enc' for encryption.
-x, --proxyFilename <value>	Sets the proxy file to use as client credential.
-z, --authorization <type>	Specifies authorization type. type can be 'self' , 'host' , 'none' , or a string specifying the expected identity of the remote party.
-t, --timeout <timeout>	Specifies client timeout (in seconds). The client will wait maximum of the timeout value for a response from the server before returning an error. By default the timeout value is 10 minutes.

Examples:

```
$ wsrif-query -s https://127.0.0.1:8443/wsrif/services/DefaultIndexService \
  "count(//*[local-name()='Entry'])"
```

```
$ wsrif-query -s https://127.0.0.1:8443/wsrif/services/DefaultIndexService \
  "number(//*[local-name()='GLUECE']/glue:ComputingElement/glue:State/@glue:FreeCPUs)=0"
```

```
$ wsrif-query -s http://localhost:8080/wsrif/services/ContainerRegistryService \
  "/*/*/*/*[local-name()='Address']"
```

Name

`wsrf-get-property --` Gets values of a single resource property

`wsrf-get-property`

Tool description

Gets a single resource property from a resource.

Command syntax

`wsrf-get-property [options] <property>`

The `<property>` is a QName of the resource property in the string form: `{namespaceURI}localPart`.

DRAFT

Table 10. Common options

-h, --help	Displays help information about the command.
-d, --debug	Enables debug mode. For example, full stack traces of errors will be displayed.
-e, --eprFile <file>	Specifies an <i>XML</i> file that contains the <i>WS-Addressing</i> endpoint reference.
-s, --service <url>	Specifies the service URL.
-k, --key <name value>	Specifies the resource key. The name is the QName of the resource key in the string form: {namespaceURI}localPart , while the value is the simple value of the key. For complex keys, use the --eprFile option. Example: <pre>-k "{http://www.globus.org}MyKey" 123</pre>
-f, --descriptor <file>	Specifies a client security descriptor. Overrides all other security settings.
-a, --anonymous	Enables anonymous authentication. Only supported with transport security or the GSI Secure Conversation authentication mechanism.
-g, --delegation <mode>	Enables delegation. mode can be either 'limited' or 'full' . Only supported with the GSI Secure Conversation authentication mechanism.
-l, --contextLifetime <value>	Sets the lifetime of the client security context. value is in milliseconds. Only supported with the GSI Secure Conversation authentication mechanism.
-m, --securityMech <type>	Specifies the authentication mechanism. type can be 'msg' for GSI Secure Message, or 'conv' for GSI Secure Conversation.
-c, --serverCertificate <file>	Specifies the server's <i>certificate</i> file used for encryption. Only needed for the GSI Secure Message authentication mechanism.
-p, --protection <type>	Specifies the protection level. type can be 'sig' for signature or 'enc' for encryption.
-x, --proxyFilename <value>	Sets the proxy file to use as client credential.
-z, --authorization <type>	Specifies authorization type. type can be 'self' , 'host' , 'none' , or a string specifying the expected identity of the remote party.
-t, --timeout <timeout>	Specifies client timeout (in seconds). The client will wait maximum of the timeout value for a response from the server before returning an error. By default the timeout value is 10 minutes.

Example:

```
$ wsrif-get-property -s http://localhost:8080/wsrif/services/CounterService \ -k
  "{http://counter.com}CounterKey" 123 \
  "{http://docs.oasis-open.org/wsrif/2004/06/wsrif-WS-ResourceLifetime-1.2-draft-01.xsd}Cu
```

Name

wsrf-get-properties -- Gets values of multiple resource properties

wsrf-get-properties

Tool description

Gets multiple resource properties from a resource.

Command syntax

```
wsrf-get-properties [options] <property1> [<property2>...  
  <propertyN>]
```

Each **<propertyN>** is a QName of the resource property in the string form: **{namespaceURI}localPart**.

DRAFT

Table 11. Common options

-h, --help	Displays help information about the command.
-d, --debug	Enables debug mode. For example, full stack traces of errors will be displayed.
-e, --eprFile <file>	Specifies an <i>XML</i> file that contains the <i>WS-Addressing</i> endpoint reference.
-s, --service <url>	Specifies the service URL.
-k, --key <name value>	Specifies the resource key. The name is the QName of the resource key in the string form: {namespaceURI}localPart , while the value is the simple value of the key. For complex keys, use the --eprFile option. Example: <pre>-k "{http://www.globus.org}MyKey" 123</pre>
-f, --descriptor <file>	Specifies a client security descriptor. Overrides all other security settings.
-a, --anonymous	Enables anonymous authentication. Only supported with transport security or the GSI Secure Conversation authentication mechanism.
-g, --delegation <mode>	Enables delegation. mode can be either 'limited' or 'full' . Only supported with the GSI Secure Conversation authentication mechanism.
-l, --contextLifetime <value>	Sets the lifetime of the client security context. value is in milliseconds. Only supported with the GSI Secure Conversation authentication mechanism.
-m, --securityMech <type>	Specifies the authentication mechanism. type can be 'msg' for GSI Secure Message, or 'conv' for GSI Secure Conversation.
-c, --serverCertificate <file>	Specifies the server's <i>certificate</i> file used for encryption. Only needed for the GSI Secure Message authentication mechanism.
-p, --protection <type>	Specifies the protection level. type can be 'sig' for signature or 'enc' for encryption.
-x, --proxyFilename <value>	Sets the proxy file to use as client credential.
-z, --authorization <type>	Specifies authorization type. type can be 'self' , 'host' , 'none' , or a string specifying the expected identity of the remote party.
-t, --timeout <timeout>	Specifies client timeout (in seconds). The client will wait maximum of the timeout value for a response from the server before returning an error. By default the timeout value is 10 minutes.

Example:

```
$ wsrif-get-properties -s http://localhost:8080/wsrif/services/CounterService \ -k
  "{http://counter.com}CounterKey" 123 \
  "{http://docs.oasis-open.org/wsrif/2004/06/wsrif-WS-ResourceLifetime-1.2-draft-01.xsd}Cu
  \
  "{http://docs.oasis-open.org/wsrif/2004/06/wsrif-WS-ResourceLifetime-1.2-draft-01.xsd}Te
```

Name

`wsr-insert-property` -- Inserts values into a resource property

`wsr-insert-property`

Tool description

Inserts a property into the resource property document of a resource.

Command syntax

```
wsr-insert-property [options] <propertyValueFile>
```

The **<propertyValueFile>** is an XML file that contains the value of the resource property. The QName of the property is the outer most element.

DRAFT

Table 12. Common options

-h, --help	Displays help information about the command.
-d, --debug	Enables debug mode. For example, full stack traces of errors will be displayed.
-e, --eprFile <file>	Specifies an <i>XML</i> file that contains the <i>WS-Addressing</i> endpoint reference.
-s, --service <url>	Specifies the service URL.
-k, --key <name value>	Specifies the resource key. The name is the QName of the resource key in the string form: {namespaceURI}localPart , while the value is the simple value of the key. For complex keys, use the --eprFile option. Example: <pre>-k "{http://www.globus.org}MyKey" 123</pre>
-f, --descriptor <file>	Specifies a client security descriptor. Overrides all other security settings.
-a, --anonymous	Enables anonymous authentication. Only supported with transport security or the GSI Secure Conversation authentication mechanism.
-g, --delegation <mode>	Enables delegation. mode can be either 'limited' or 'full' . Only supported with the GSI Secure Conversation authentication mechanism.
-l, --contextLifetime <value>	Sets the lifetime of the client security context. value is in milliseconds. Only supported with the GSI Secure Conversation authentication mechanism.
-m, --securityMech <type>	Specifies the authentication mechanism. type can be 'msg' for GSI Secure Message, or 'conv' for GSI Secure Conversation.
-c, --serverCertificate <file>	Specifies the server's <i>certificate</i> file used for encryption. Only needed for the GSI Secure Message authentication mechanism.
-p, --protection <type>	Specifies the protection level. type can be 'sig' for signature or 'enc' for encryption.
-x, --proxyFilename <value>	Sets the proxy file to use as client credential.
-z, --authorization <type>	Specifies authorization type. type can be 'self' , 'host' , 'none' , or a string specifying the expected identity of the remote party.
-t, --timeout <timeout>	Specifies client timeout (in seconds). The client will wait maximum of the timeout value for a response from the server before returning an error. By default the timeout value is 10 minutes.

Example: Contents of **in.xml**:

```
<doc> <ns1:foo xmlns:ns1="http://widgets.com"> Value1
  </ns1:foo> <ns1:foo xmlns:ns1="http://widgets.com"> Value2
</ns1:foo> </doc>
```

```
$ wsrf-insert-property -s http://localhost:8080/wsrf/services/WidgetService \ -k
  "{http://www.globus.org/namespaces/2004/06/core}WidgetKey" 123 \ in.xml
```

Name

wsrf-update-property -- Updates value of a resource property

wsrf-update-property

Tool description

Updates the property value in the resource property document of a resource.

Command syntax

```
wsrf-update-property [options] <propertyValueFile>
```

The **<propertyValueFile>** is an XML file that contains the value of the resource property. The QName of the property is the outermost element.

DRAFT

Table 13. Common options

-h, --help	Displays help information about the command.
-d, --debug	Enables debug mode. For example, full stack traces of errors will be displayed.
-e, --eprFile <file>	Specifies an <i>XML</i> file that contains the <i>WS-Addressing</i> endpoint reference.
-s, --service <url>	Specifies the service URL.
-k, --key <name value>	Specifies the resource key. The name is the QName of the resource key in the string form: {namespaceURI}localPart , while the value is the simple value of the key. For complex keys, use the --eprFile option. Example: <pre>-k "{http://www.globus.org}MyKey" 123</pre>
-f, --descriptor <file>	Specifies a client security descriptor. Overrides all other security settings.
-a, --anonymous	Enables anonymous authentication. Only supported with transport security or the GSI Secure Conversation authentication mechanism.
-g, --delegation <mode>	Enables delegation. mode can be either 'limited' or 'full' . Only supported with the GSI Secure Conversation authentication mechanism.
-l, --contextLifetime <value>	Sets the lifetime of the client security context. value is in milliseconds. Only supported with the GSI Secure Conversation authentication mechanism.
-m, --securityMech <type>	Specifies the authentication mechanism. type can be 'msg' for GSI Secure Message, or 'conv' for GSI Secure Conversation.
-c, --serverCertificate <file>	Specifies the server's <i>certificate</i> file used for encryption. Only needed for the GSI Secure Message authentication mechanism.
-p, --protection <type>	Specifies the protection level. type can be 'sig' for signature or 'enc' for encryption.
-x, --proxyFilename <value>	Sets the proxy file to use as client credential.
-z, --authorization <type>	Specifies authorization type. type can be 'self' , 'host' , 'none' , or a string specifying the expected identity of the remote party.
-t, --timeout <timeout>	Specifies client timeout (in seconds). The client will wait maximum of the timeout value for a response from the server before returning an error. By default the timeout value is 10 minutes.

Example: Contents of **in.xml**:

```
<doc> <ns1:foo xmlns:ns1="http://widgets.com"> Value
  </ns1:foo> </doc>
```

```
$ wsrfe-update-property -s http://localhost:8080/wsrfe/services/WidgetService \ -k
  "{http://www.globus.org/namespaces/2004/06/core}WidgetKey" 123 \ in.xml
```

Name

`wsrf-delete-property --` Deletes a resource property

`wsrf-delete-property`

Tool description

Deletes a resource property from the resource property document of a resource.

Command syntax

```
wsrf-delete-property [options] <property>
```

The `<property>` is a QName of the resource property in the string form: `{namespaceURI}localPart`.

DRAFT

Table 14. Common options

-h, --help	Displays help information about the command.
-d, --debug	Enables debug mode. For example, full stack traces of errors will be displayed.
-e, --eprFile <file>	Specifies an <i>XML</i> file that contains the <i>WS-Addressing</i> endpoint reference.
-s, --service <url>	Specifies the service URL.
-k, --key <name value>	Specifies the resource key. The name is the QName of the resource key in the string form: {namespaceURI}localPart , while the value is the simple value of the key. For complex keys, use the --eprFile option. Example: <pre>-k "{http://www.globus.org}MyKey" 123</pre>
-f, --descriptor <file>	Specifies a client security descriptor. Overrides all other security settings.
-a, --anonymous	Enables anonymous authentication. Only supported with transport security or the GSI Secure Conversation authentication mechanism.
-g, --delegation <mode>	Enables delegation. mode can be either 'limited' or 'full' . Only supported with the GSI Secure Conversation authentication mechanism.
-l, --contextLifetime <value>	Sets the lifetime of the client security context. value is in milliseconds. Only supported with the GSI Secure Conversation authentication mechanism.
-m, --securityMech <type>	Specifies the authentication mechanism. type can be 'msg' for GSI Secure Message, or 'conv' for GSI Secure Conversation.
-c, --serverCertificate <file>	Specifies the server's <i>certificate</i> file used for encryption. Only needed for the GSI Secure Message authentication mechanism.
-p, --protection <type>	Specifies the protection level. type can be 'sig' for signature or 'enc' for encryption.
-x, --proxyFilename <value>	Sets the proxy file to use as client credential.
-z, --authorization <type>	Specifies authorization type. type can be 'self' , 'host' , 'none' , or a string specifying the expected identity of the remote party.
-t, --timeout <timeout>	Specifies client timeout (in seconds). The client will wait maximum of the timeout value for a response from the server before returning an error. By default the timeout value is 10 minutes.

Example:

```
$ wsrfr-delete-property -s http://localhost:8080/wsrfr/services/WidgetService \ -k
  "{http://www.globus.org/namespaces/2004/06/core}WidgetKey" 123 \
  "{http://widgets.com}foo"
```

Name

`wsn-get-current-message --` Gets a current message associated with a topic

`wsn-get-current-message`

Tool description

Gets the current message associated with the specified topic.

Command syntax

`wsn-get-current-message [options] <topic>`

The `<topic>` is a QName of the resource property in the string form: `{namespaceURI}localPart`.

DRAFT

Table 15. Common options

-h, --help	Displays help information about the command.
-d, --debug	Enables debug mode. For example, full stack traces of errors will be displayed.
-e, --eprFile <file>	Specifies an <i>XML</i> file that contains the <i>WS-Addressing</i> endpoint reference.
-s, --service <url>	Specifies the service URL.
-k, --key <name value>	Specifies the resource key. The name is the QName of the resource key in the string form: {namespaceURI}localPart , while the value is the simple value of the key. For complex keys, use the --eprFile option. Example: <pre>-k "{http://www.globus.org}MyKey" 123</pre>
-f, --descriptor <file>	Specifies a client security descriptor. Overrides all other security settings.
-a, --anonymous	Enables anonymous authentication. Only supported with transport security or the GSI Secure Conversation authentication mechanism.
-g, --delegation <mode>	Enables delegation. mode can be either 'limited' or 'full' . Only supported with the GSI Secure Conversation authentication mechanism.
-l, --contextLifetime <value>	Sets the lifetime of the client security context. value is in milliseconds. Only supported with the GSI Secure Conversation authentication mechanism.
-m, --securityMech <type>	Specifies the authentication mechanism. type can be 'msg' for GSI Secure Message, or 'conv' for GSI Secure Conversation.
-c, --serverCertificate <file>	Specifies the server's <i>certificate</i> file used for encryption. Only needed for the GSI Secure Message authentication mechanism.
-p, --protection <type>	Specifies the protection level. type can be 'sig' for signature or 'enc' for encryption.
-x, --proxyFilename <value>	Sets the proxy file to use as client credential.
-z, --authorization <type>	Specifies authorization type. type can be 'self' , 'host' , 'none' , or a string specifying the expected identity of the remote party.
-t, --timeout <timeout>	Specifies client timeout (in seconds). The client will wait maximum of the timeout value for a response from the server before returning an error. By default the timeout value is 10 minutes.

Example:

```
$ wsn-get-current-message -s
  http://localhost:8080/wsrf/services/CounterService \ -k "{http://counter.com}CounterK
  "{http://counter.com}Value"
```

Name

`wsn-pause-subscription --` Pauses a subscription

`wsn-pause-subscription`

Tool description

Pauses a subscription (notifications on that subscription will not be sent out until it is resumed).

Command syntax

`wsn-pause-subscription [options]`

DRAFT

Table 16. Common options

-h, --help	Displays help information about the command.
-d, --debug	Enables debug mode. For example, full stack traces of errors will be displayed.
-e, --eprFile <file>	Specifies an <i>XML</i> file that contains the <i>WS-Addressing</i> endpoint reference.
-s, --service <url>	Specifies the service URL.
-k, --key <name value>	Specifies the resource key. The name is the QName of the resource key in the string form: {namespaceURI}localPart , while the value is the simple value of the key. For complex keys, use the --eprFile option. Example: <pre>-k "{http://www.globus.org}MyKey" 123</pre>
-f, --descriptor <file>	Specifies a client security descriptor. Overrides all other security settings.
-a, --anonymous	Enables anonymous authentication. Only supported with transport security or the GSI Secure Conversation authentication mechanism.
-g, --delegation <mode>	Enables delegation. mode can be either 'limited' or 'full' . Only supported with the GSI Secure Conversation authentication mechanism.
-l, --contextLifetime <value>	Sets the lifetime of the client security context. value is in milliseconds. Only supported with the GSI Secure Conversation authentication mechanism.
-m, --securityMech <type>	Specifies the authentication mechanism. type can be 'msg' for GSI Secure Message, or 'conv' for GSI Secure Conversation.
-c, --serverCertificate <file>	Specifies the server's <i>certificate</i> file used for encryption. Only needed for the GSI Secure Message authentication mechanism.
-p, --protection <type>	Specifies the protection level. type can be 'sig' for signature or 'enc' for encryption.
-x, --proxyFilename <value>	Sets the proxy file to use as client credential.
-z, --authorization <type>	Specifies authorization type. type can be 'self' , 'host' , 'none' , or a string specifying the expected identity of the remote party.
-t, --timeout <timeout>	Specifies client timeout (in seconds). The client will wait maximum of the timeout value for a response from the server before returning an error. By default the timeout value is 10 minutes.

Example:

```
$ wsn-pause-subscription -s
  http://localhost:8080/wsrf/services/SubscriptionManagerService \ -k
  "{http://www.globus.org/namespaces/2004/06/core}acc271c0-4df9-11d9-ab19-87da3bc7cf28"
```

Name

`wsn-resume-subscription --` Resumes a subscription

`wsn-resume-subscription`

Tool description

Resumes a subscription (notifications on that subscription will be sent out again).

Command syntax

`wsn-resume-subscription [options]`

DRAFT

Table 17. Common options

-h, --help	Displays help information about the command.
-d, --debug	Enables debug mode. For example, full stack traces of errors will be displayed.
-e, --eprFile <file>	Specifies an <i>XML</i> file that contains the <i>WS-Addressing</i> endpoint reference.
-s, --service <url>	Specifies the service URL.
-k, --key <name value>	Specifies the resource key. The name is the QName of the resource key in the string form: {namespaceURI}localPart , while the value is the simple value of the key. For complex keys, use the --eprFile option. Example: <pre>-k "{http://www.globus.org}MyKey" 123</pre>
-f, --descriptor <file>	Specifies a client security descriptor. Overrides all other security settings.
-a, --anonymous	Enables anonymous authentication. Only supported with transport security or the GSI Secure Conversation authentication mechanism.
-g, --delegation <mode>	Enables delegation. mode can be either 'limited' or 'full' . Only supported with the GSI Secure Conversation authentication mechanism.
-l, --contextLifetime <value>	Sets the lifetime of the client security context. value is in milliseconds. Only supported with the GSI Secure Conversation authentication mechanism.
-m, --securityMech <type>	Specifies the authentication mechanism. type can be 'msg' for GSI Secure Message, or 'conv' for GSI Secure Conversation.
-c, --serverCertificate <file>	Specifies the server's <i>certificate</i> file used for encryption. Only needed for the GSI Secure Message authentication mechanism.
-p, --protection <type>	Specifies the protection level. type can be 'sig' for signature or 'enc' for encryption.
-x, --proxyFilename <value>	Sets the proxy file to use as client credential.
-z, --authorization <type>	Specifies authorization type. type can be 'self' , 'host' , 'none' , or a string specifying the expected identity of the remote party.
-t, --timeout <timeout>	Specifies client timeout (in seconds). The client will wait maximum of the timeout value for a response from the server before returning an error. By default the timeout value is 10 minutes.

Example:

```
$ wsn-resume-subscription -s
  http://localhost:8080/wsrf/services/SubscriptionManagerService \ -k
  "{http://www.globus.org/namespaces/2004/06/core}acc271c0-4df9-11d9-ab19-87da3bc7cf28"
```

Name

wsn-subscribe -- Subscribes to a topic

wsn-subscribe

Tool description

Subscribes to a topic.

Command syntax

```
wsn-subscribe [options] <topic>
```

The <topic> is a QName of the resource property in the string form: **{namespaceURI}localPart**.

The following are subscribe-specific options in addition to the common options:

Table 18. Command-specific options

-r, --resDescFile <file>	Specifies a file containing a resource security descriptor for the notification consumer resource.
-b, --subEpr <file>	Specifies a file to which the subscription resource EPR will be saved.

Table 19. Common options

-h, --help	Displays help information about the command.
-d, --debug	Enables debug mode. For example, full stack traces of errors will be displayed.
-e, --eprFile <file>	Specifies an <i>XML</i> file that contains the <i>WS-Addressing</i> endpoint reference.
-s, --service <url>	Specifies the service URL.
-k, --key <name value>	Specifies the resource key. The name is the QName of the resource key in the string form: {namespaceURI}localPart , while the value is the simple value of the key. For complex keys, use the --eprFile option. Example: <pre>-k "{http://www.globus.org}MyKey" 123</pre>
-f, --descriptor <file>	Specifies a client security descriptor. Overrides all other security settings.
-a, --anonymous	Enables anonymous authentication. Only supported with transport security or the GSI Secure Conversation authentication mechanism.
-g, --delegation <mode>	Enables delegation. mode can be either 'limited' or 'full' . Only supported with the GSI Secure Conversation authentication mechanism.
-l, --contextLifetime <value>	Sets the lifetime of the client security context. value is in milliseconds. Only supported with the GSI Secure Conversation authentication mechanism.
-m, --securityMech <type>	Specifies the authentication mechanism. type can be 'msg' for GSI Secure Message, or 'conv' for GSI Secure Conversation.
-c, --serverCertificate <file>	Specifies the server's <i>certificate</i> file used for encryption. Only needed for the GSI Secure Message authentication mechanism.
-p, --protection <type>	Specifies the protection level. type can be 'sig' for signature or 'enc' for encryption.
-x, --proxyFilename <value>	Sets the proxy file to use as client credential.
-z, --authorization <type>	Specifies authorization type. type can be 'self' , 'host' , 'none' , or a string specifying the expected identity of the remote party.
-t, --timeout <timeout>	Specifies client timeout (in seconds). The client will wait maximum of the timeout value for a response from the server before returning an error. By default the timeout value is 10 minutes.

Example:

```
$ wsn-subscribe -s http://localhost:8080/wsrf/services/CounterService \ -k
  "{http://counter.com}CounterKey" 123 \ "{http://counter.com}Value"
```

Name

globus-deploy-gar -- Deploys a GAR file (locally)

globus-deploy-gar

Tool description

Deploys a GAR file (locally) into Java WS Core or Apache Tomcat container.

Command syntax

```
globus-deploy-gar [options] <gar.file>
```

The **<gar.file>** is the path to the GAR file to be deployed.

Table 20. Options

-help	Displays help information about the command.
-debug	Enables debug mode.
-verbose	Enables verbose mode.
-backup	Creates backup of existing configuration files.
-overwrite	Overwrite existing deployment.
-profile <name>	Specifies the profile name under which the configuration files in the GAR will be deployed. Please see "Configuration Profiles" under Configuring Java WS Core for details.
-tomcat <dir>	Deploys a GAR file to Apache Tomcat. The <dir> argument must point to the Tomcat installation directory. <i>Note:</i> Java WS Core must be already deployed in Tomcat. Please see Deploying into Tomcat section for details.
-D<property>=<value>	Passes arbitrary property-value pairs. See below for the list of currently supported properties .

Table 21. Supported property-value pairs

-Dall.scripts=true	Causes Windows and Unix launcher scripts to be generated.
-DdoValidation=false	Turns off automatic validation of service configuration files.



Note

Since GT 4.2, **globus-deploy-gar** command will NOT overwrite the existing deployment unless **-overwrite** option is specified. It is recommended to undeploy the existing deployment first. The container must be off to deploy a GAR file.

Example 1:

```
$ globus-deploy-gar /tmp/gars/globus_wsrf_core_samples_counter.gar
```

The above command will deploy the `globus_wsrf_core_samples_counter.gar` into Java WS Core installation directory. The above command invokes the `deployGar` task in the `build-packages.xml` Ant build file. The above example is equivalent to running:

```
$ ant -f $GLOBUS_LOCATION/share/globus_wsrf_common/build-packages.xml deployGar \  
-Dgar.name=/tmp/gars/globus_wsrf_core_samples_counter.gar
```

The profile name can be passed using the **-Dprofile** Ant option. To enable back up of the existing configuration files add the **-DcreateBackup=true** Ant option. Make sure to use the *absolute* path name for the gar file when using Ant directly.

Example II:

```
$ globus-deploy-gar -tomcat /soft/tomcat-5.5.20 \  
/tmp/gars/globus_wsrf_core_samples_counter.gar
```

The above command will deploy the `globus_wsrf_core_samples_counter.gar` into Apache Tomcat. The above command invokes the `deployGar` task in the `tomcat-service.xml` Ant build file. The above example is equivalent to running:

```
$ ant -f $GLOBUS_LOCATION/share/globus_wsrf_common/tomcat/tomcat-service.xml deployGar \  
-Dgar.name=/tmp/gars/globus_wsrf_core_samples_counter.gar \ -Dtomcat.dir=/soft/tomcat-
```

By default the GAR file will be deployed under the "wsrf" web application. To specify a different web application name use **-Dwebapp.name=<name>** option.

Name

globus-undeploy-gar -- Undeploys a GAR file (locally)

globus-undeploy-gar

Tool description

Undeploys a GAR file (locally) from Java WS Core or Apache Tomcat container.

Command syntax

```
globus-undeploy-gar [options] <gar.id>
```

The <gar.id> is the base name of the GAR file without the **.gar** extension to undeploy. For example if the GAR file is "foo.gar", then the GAR id is "foo". The directory names under **\$GLOBUS_LOCATION/etc/globus_packages/** are the GAR ids of the undeployable items.

Table 22. Options

-help	Displays help information about the command.
-debug	Enables debug mode.
-verbose	Enables verbose mode.
-tomcat <dir>	Undeploy a GAR file from Apache Tomcat. The <dir> argument must point to the Tomcat installation directory.
-D<property>=<value>	Passes arbitrary property-value pairs.



Note

The container must be off to undeploy a GAR file.

Example I:

```
$ globus-undeploy-gar globus_wsrf_core_samples_counter
```

The above command will undeploy globus_wsrf_core_samples_counter GAR from Java WS Core installation directory. The above command invokes the **undeployGar** task in the build-packages.xml Ant build file. The above example is equivalent to running:

```
$ ant -f $GLOBUS_LOCATION/share/globus_wsrf_common/build-packages.xml undeployGar \
-Dgar.id=globus_wsrf_core_samples_counter
```

Example II:

```
$ globus-undeploy-gar -tomcat /soft/tomcat-5.5.20 \ globus_wsrf_core_samples_counter
```

The above command will undeploy globus_wsrf_core_samples_counter GAR from Apache Tomcat. The above command invokes the **undeployGar** task in the tomcat-service.xml Ant build file. The above example is equivalent to running:

```
$ ant -f $GLOBUS_LOCATION/share/globus_wsrf_common/tomcat/tomcat-service.xml undeployGar \
-Dgar.id=globus_wsrf_core_samples_counter \ -Dtomcat.dir=/soft/tomcat-5.5.20
```

By default the GAR file will be undeployed under the "wsrf" web application. To specify a different web application name use **-Dwebapp.name=<name>** option.

DRAFT

Name

globus-check-environment -- Displays component version information and validates JVM version.

globus-check-environment

Tool description

Displays component version information and validates the JVM version. This tool is primarily used for debugging purposes.

DRAFT

Name

`globus-check-remote-environment --` Displays remote component version information.

`globus-check-remote-environment`

Tool description

Displays remote component version information.

Command syntax

```
globus-check-environment [-help] -s endpoint -z authz
```

Table 23. Options

-help	Displays help information about the command.
-s endpoint	Remote endpoint to print version information about. It should be of the format protocol://host:port, example https://localhost:8443.
-z authz	Sets authorization, can be 'self', 'host', 'hostOrSelf' or 'none' or a string specifying the expected identity of the remote party. Defaults to no authorization.

Name

`globus-update-client-config --` Merges `client-config.wsdd` files from deployed modules into the global `client-config.wsdd` configuration file

```
globus-update-client-config
```

Tool description

Merges multiple `client-config.wsdd` files from deployed modules into the global configuration file. Scans each `$GLOBUS_LOCATION/etc/<modulename>/client-config.wsdd` and merges the contents into `$GLOBUS_LOCATION/client-config.wsdd`. This tool is primarily intended for use by administrators and automation tools to facilitate the adding and removing of module specific type-mapping and/or other client-side configuration from the global `client-config.wsdd` file used by the Globus installation.

Command syntax

```
globus-update-client-config [<filename>]
```

Table 24. Options

<filename>	Optional argument that specifies an alternate path to write the result <code>client-config.wsdd</code> file. By default, running the program with no arguments will write the file to <code>\$GLOBUS_LOCATION/client-config.wsdd</code>
-------------------------	---

Name

globus-validate-descriptors -- Validate configuration files of all services

globus-validate-descriptors

Tool description

Validates the Web Services Deployment Descriptor (.wsdd) files, JNDI configuration files (jndi-config.xml), and security descriptors for all services.

Command syntax

```
globus-validate-descriptors [options]
```

Table 25. Options

-help	Displays help information about the command.
-debug	Enables debug mode.
-verbose	Enables verbose mode.
-D<property>=<value>	Passes arbitrary property-value pairs.

Name

globus-reload-container -- Reinitializes standalone container

globus-reload-container

Tool description

Invokes the **reload()** operation on the **DeployService** running in the remote container. It tells the container to reinitialize all of its services, re-read its and service configuration files, etc. For example, the administrator can change the security descriptor of a service and then use the **globus-reload-container** command to force the container to load the updated configuration without restarting the container.

By default the **DeployService** is configured to perform **self** authorization and therefore the **globus-reload-container** must be executed with the same credentials as the container it is running with. Alternatively, the service can be configured with a gridmap file to allow a subset of users (with their own credentials) to invoke the service (please see the service security deployment descriptor for details).



Note

This command only works with the standalone container. Please see the [Java WS Core Dynamic Deploy Design Document](#) for more information.

Command syntax

globus-reload-container [options]

Table 26. Common options

-h, --help	Displays help information about the command.
-d, --debug	Enables debug mode. For example, full stack traces of errors will be displayed.
-e, --eprFile <file>	Specifies an <i>XML</i> file that contains the <i>WS-Addressing</i> endpoint reference.
-s, --service <url>	Specifies the service URL.
-k, --key <name value>	Specifies the resource key. The name is the QName of the resource key in the string form: {namespaceURI}localPart , while the value is the simple value of the key. For complex keys, use the --eprFile option. Example: <pre>-k "{http://www.globus.org}MyKey" 123</pre>
-f, --descriptor <file>	Specifies a client security descriptor. Overrides all other security settings.
-a, --anonymous	Enables anonymous authentication. Only supported with transport security or the GSI Secure Conversation authentication mechanism.
-g, --delegation <mode>	Enables delegation. mode can be either 'limited' or 'full' . Only supported with the GSI Secure Conversation authentication mechanism.
-l, --contextLifetime <value>	Sets the lifetime of the client security context. value is in milliseconds. Only supported with the GSI Secure Conversation authentication mechanism.
-m, --securityMech <type>	Specifies the authentication mechanism. type can be 'msg' for GSI Secure Message, or 'conv' for GSI Secure Conversation.
-c, --serverCertificate <file>	Specifies the server's <i>certificate</i> file used for encryption. Only needed for the GSI Secure Message authentication mechanism.
-p, --protection <type>	Specifies the protection level. type can be 'sig' for signature or 'enc' for encryption.
-x, --proxyFilename <value>	Sets the proxy file to use as client credential.
-z, --authorization <type>	Specifies authorization type. type can be 'self' , 'host' , 'none' , or a string specifying the expected identity of the remote party.
-t, --timeout <timeout>	Specifies client timeout (in seconds). The client will wait maximum of the timeout value for a response from the server before returning an error. By default the timeout value is 10 minutes.

Example:

```
$ globus-reload-container
```

Name

`globus-remote-undeploy-gar --` Undeploys a GAR file (remotely)

`globus-remote-undeploy-gar`

Tool description

The **globus-remote-undeploy-gar** command undeploys a GAR file remotely. It invokes the **undeploy()** operation on the **DeployService** running in the remote container. It works just like the `globus-undeploy-gar` command but the GAR file is undeployed remotely.

By default the **DeployService** is configured to perform **self** authorization and therefore the **globus-remote-undeploy-gar** must be executed with the same credentials as the container it is running with. Alternatively, the service can be configured with a gridmap file to allow a subset of users (with their own credentials) to invoke the service (please see the service security deployment descriptor for details).



Note

This command only works with the standalone container. Please see the [Java WS Core Dynamic Deploy Design Document](#) for more information.

Command syntax

```
globus-remote-undeploy-gar [options] <gar.id>
```

The `<gar.id>` is the base name of the GAR file without the `.gar` extension to undeploy. For example if the GAR file is `"foo.gar"`, then the GAR id is `"foo"`.

Table 27. Common options

-h, --help	Displays help information about the command.
-d, --debug	Enables debug mode. For example, full stack traces of errors will be displayed.
-e, --eprFile <file>	Specifies an <i>XML</i> file that contains the <i>WS-Addressing</i> endpoint reference.
-s, --service <url>	Specifies the service URL.
-k, --key <name value>	Specifies the resource key. The name is the QName of the resource key in the string form: {namespaceURI}localPart , while the value is the simple value of the key. For complex keys, use the --eprFile option. Example: <pre>-k "{http://www.globus.org}MyKey" 123</pre>
-f, --descriptor <file>	Specifies a client security descriptor. Overrides all other security settings.
-a, --anonymous	Enables anonymous authentication. Only supported with transport security or the GSI Secure Conversation authentication mechanism.
-g, --delegation <mode>	Enables delegation. mode can be either 'limited' or 'full' . Only supported with the GSI Secure Conversation authentication mechanism.
-l, --contextLifetime <value>	Sets the lifetime of the client security context. value is in milliseconds. Only supported with the GSI Secure Conversation authentication mechanism.
-m, --securityMech <type>	Specifies the authentication mechanism. type can be 'msg' for GSI Secure Message, or 'conv' for GSI Secure Conversation.
-c, --serverCertificate <file>	Specifies the server's <i>certificate</i> file used for encryption. Only needed for the GSI Secure Message authentication mechanism.
-p, --protection <type>	Specifies the protection level. type can be 'sig' for signature or 'enc' for encryption.
-x, --proxyFilename <value>	Sets the proxy file to use as client credential.
-z, --authorization <type>	Specifies authorization type. type can be 'self' , 'host' , 'none' , or a string specifying the expected identity of the remote party.
-t, --timeout <timeout>	Specifies client timeout (in seconds). The client will wait maximum of the timeout value for a response from the server before returning an error. By default the timeout value is 10 minutes.

Example:

```
$ globus-remote-undeploy-gar globus_wsrf_core_samples_counter
```

To see what GAR files can be undeployed on the remote container run the following query on the **DeployService**, for example:

```
$ wsrf-query -z hostSelf -s https://127.0.0.1:8443/wsrf/services/DeployService
```

Name

globus-remote-deploy-gar -- Deploys a GAR file (remotely)

globus-remote-deploy-gar

Tool description

The **globus-remote-deploy-gar** command deploys a GAR file remotely. It first transfers the GAR file to the **DeployService** running in the remote container and then it deploys it using the **deploy()** operation of the service (the tool can also perform these two operations separately).

By default the **DeployService** is configured to perform **self** authorization and therefore the **globus-remote-deploy-gar** must be executed with the same credentials as the container it is running with. Alternatively, the service can be configured with a gridmap file to allow a subset of users (with their own credentials) to invoke the service (please see the service security deployment descriptor for details).



Note

This command only works with the standalone container. Please see the [Java WS Core Dynamic Deploy Design Document](#) for more information.

Command syntax

```
globus-remote-deploy-gar [options] <gar>
```

The **<gar>** can be either an URL or a file location. If a file location is passed to the tool, it will transfer the file to the service via SOAP with Attachments (the **upload()** function) using the **MTOM** format. If an URL is passed, the tool will call the **download()** function of the service, and let the service download the GAR file.

The following are command-specific options in addition to the common options:

Table 28. Command-specific options

-n, --transfer	Transfer GAR file only.
-y, --deploy	Deploy GAR file only (assumes the GAR is already transferred to the DeployService).
-o, --overwrite	Overwrite existing deployment.
-b, --backup	Creates backup of existing configuration files

Table 29. Common options

-h, --help	Displays help information about the command.
-d, --debug	Enables debug mode. For example, full stack traces of errors will be displayed.
-e, --eprFile <file>	Specifies an <i>XML</i> file that contains the <i>WS-Addressing</i> endpoint reference.
-s, --service <url>	Specifies the service URL.
-k, --key <name value>	Specifies the resource key. The name is the QName of the resource key in the string form: {namespaceURI}localPart , while the value is the simple value of the key. For complex keys, use the --eprFile option. Example: <pre>-k "{http://www.globus.org}MyKey" 123</pre>
-f, --descriptor <file>	Specifies a client security descriptor. Overrides all other security settings.
-a, --anonymous	Enables anonymous authentication. Only supported with transport security or the GSI Secure Conversation authentication mechanism.
-g, --delegation <mode>	Enables delegation. mode can be either 'limited' or 'full' . Only supported with the GSI Secure Conversation authentication mechanism.
-l, --contextLifetime <value>	Sets the lifetime of the client security context. value is in milliseconds. Only supported with the GSI Secure Conversation authentication mechanism.
-m, --securityMech <type>	Specifies the authentication mechanism. type can be 'msg' for GSI Secure Message, or 'conv' for GSI Secure Conversation.
-c, --serverCertificate <file>	Specifies the server's <i>certificate</i> file used for encryption. Only needed for the GSI Secure Message authentication mechanism.
-p, --protection <type>	Specifies the protection level. type can be 'sig' for signature or 'enc' for encryption.
-x, --proxyFilename <value>	Sets the proxy file to use as client credential.
-z, --authorization <type>	Specifies authorization type. type can be 'self' , 'host' , 'none' , or a string specifying the expected identity of the remote party.
-t, --timeout <timeout>	Specifies client timeout (in seconds). The client will wait maximum of the timeout value for a response from the server before returning an error. By default the timeout value is 10 minutes.

Examples:

```
$ globus-remote-deploy-gar /tmp/myService.gar
```

```
$ globus-remote-deploy-gar gsiftp://localhost/tmp/myService.gar
```

To see what GAR files have been transferred but not yet deployed on the remote container run the following query on the **DeployService**, for example:

```
$ wsrfl-query -z hostSelf -s https://127.0.0.1:8443/wsrfl/services/DeployService
```

Name

`ws-enumerate-start --` Starts an enumeration

`ws-enumerate-start`

Tool description

Creates a new enumeration context and prints it out to the console.



Note

The remote service must support the **enumerate** operation of the WS-Enumeration specification.

Command syntax

`ws-enumerate-start [options]`

DRAFT

Table 30. Common options

-h, --help	Displays help information about the command.
-d, --debug	Enables debug mode. For example, full stack traces of errors will be displayed.
-e, --eprFile <file>	Specifies an <i>XML</i> file that contains the <i>WS-Addressing</i> endpoint reference.
-s, --service <url>	Specifies the service URL.
-k, --key <name value>	Specifies the resource key. The name is the QName of the resource key in the string form: {namespaceURI}localPart , while the value is the simple value of the key. For complex keys, use the --eprFile option. Example: <pre>-k "{http://www.globus.org}MyKey" 123</pre>
-f, --descriptor <file>	Specifies a client security descriptor. Overrides all other security settings.
-a, --anonymous	Enables anonymous authentication. Only supported with transport security or the GSI Secure Conversation authentication mechanism.
-g, --delegation <mode>	Enables delegation. mode can be either 'limited' or 'full' . Only supported with the GSI Secure Conversation authentication mechanism.
-l, --contextLifetime <value>	Sets the lifetime of the client security context. value is in milliseconds. Only supported with the GSI Secure Conversation authentication mechanism.
-m, --securityMech <type>	Specifies the authentication mechanism. type can be 'msg' for GSI Secure Message, or 'conv' for GSI Secure Conversation.
-c, --serverCertificate <file>	Specifies the server's <i>certificate</i> file used for encryption. Only needed for the GSI Secure Message authentication mechanism.
-p, --protection <type>	Specifies the protection level. type can be 'sig' for signature or 'enc' for encryption.
-x, --proxyFilename <value>	Sets the proxy file to use as client credential.
-z, --authorization <type>	Specifies authorization type. type can be 'self' , 'host' , 'none' , or a string specifying the expected identity of the remote party.
-t, --timeout <timeout>	Specifies client timeout (in seconds). The client will wait maximum of the timeout value for a response from the server before returning an error. By default the timeout value is 10 minutes.

Example:

```
$ ws-enumerate-start -s http://localhost:8080/wsrfl/services/ContainerRegistryService \
> enum.context
```

The created enumeration context will be stored in the `enum.context` file which then can be passed to `ws-enumerate` and `ws-enumerate-end` command line clients.

Name

ws-enumerate -- Retrieves enumeration data

ws-enumerate

Tool description

Retrieves the next set of enumeration data and prints it out to the console.



Note

The remote service must implement the WS-Enumeration specification.

Command syntax

```
ws-enumerate [options] <enumContextFile>
```

The **<enumContextFile>** is a file that contains the enumeration context.

The following are command-specific options in addition to the common options:

Table 31. Command-specific options

-i, --items <int>	Specifies the total number of enumeration items to retrieve. The parameter value can be 'all' to retrieve the all the enumeration data. By default, only one element is retrieved.
-r, --maxCharacters <int>	Specifies the maximum number of characters (in Unicode) of the enumeration data that the client can accept at a time. By default, there is no limit on the size of the elements.
-n, --maxElements <int>	Specifies the maximum number of enumeration items to fetch at a time. By default, one element is retrieved at a time.
-o, --maxTime <int>	Specifies the maximum amount of time (in milliseconds) in which the enumeration data must be assembled. By default, there is no time limit.

Table 32. Common options

-h, --help	Displays help information about the command.
-d, --debug	Enables debug mode. For example, full stack traces of errors will be displayed.
-e, --eprFile <file>	Specifies an <i>XML</i> file that contains the <i>WS-Addressing</i> endpoint reference.
-s, --service <url>	Specifies the service URL.
-k, --key <name value>	Specifies the resource key. The name is the QName of the resource key in the string form: {namespaceURI}localPart , while the value is the simple value of the key. For complex keys, use the --eprFile option. Example: -k "{http://www.globus.org}MyKey" 123
-f, --descriptor <file>	Specifies a client security descriptor. Overrides all other security settings.
-a, --anonymous	Enables anonymous authentication. Only supported with transport security or the GSI Secure Conversation authentication mechanism.
-g, --delegation <mode>	Enables delegation. mode can be either 'limited' or 'full' . Only supported with the GSI Secure Conversation authentication mechanism.
-l, --contextLifetime <value>	Sets the lifetime of the client security context. value is in milliseconds. Only supported with the GSI Secure Conversation authentication mechanism.
-m, --securityMech <type>	Specifies the authentication mechanism. type can be 'msg' for GSI Secure Message, or 'conv' for GSI Secure Conversation.
-c, --serverCertificate <file>	Specifies the server's <i>certificate</i> file used for encryption. Only needed for the GSI Secure Message authentication mechanism.
-p, --protection <type>	Specifies the protection level. type can be 'sig' for signature or 'enc' for encryption.
-x, --proxyFilename <value>	Sets the proxy file to use as client credential.
-z, --authorization <type>	Specifies authorization type. type can be 'self' , 'host' , 'none' , or a string specifying the expected identity of the remote party.
-t, --timeout <timeout>	Specifies client timeout (in seconds). The client will wait maximum of the timeout value for a response from the server before returning an error. By default the timeout value is 10 minutes.

Example:

```
$ ws-enumerate -s http://localhost:8080/wsrf/services/ContainerRegistryService \ -i 10
-n 5 enum.context
```

This command will display 10 elements of the enumeration data obtaining 5 elements at a time from the service.

Name

`ws-enumerate-end --` Stops an enumeration

`ws-enumerate-end`

Tool description

Releases an enumeration context.



Note

The remote service must implement the WS-Enumeration specification.

Command syntax

`ws-enumerate-end [options] <enumContextFile>`

The `<enumContextFile>` is a file that contains the enumeration context.

DRAFT

Table 33. Common options

-h, --help	Displays help information about the command.
-d, --debug	Enables debug mode. For example, full stack traces of errors will be displayed.
-e, --eprFile <file>	Specifies an <i>XML</i> file that contains the <i>WS-Addressing</i> endpoint reference.
-s, --service <url>	Specifies the service URL.
-k, --key <name value>	Specifies the resource key. The name is the QName of the resource key in the string form: {namespaceURI}localPart , while the value is the simple value of the key. For complex keys, use the --eprFile option. Example: <pre>-k "{http://www.globus.org}MyKey" 123</pre>
-f, --descriptor <file>	Specifies a client security descriptor. Overrides all other security settings.
-a, --anonymous	Enables anonymous authentication. Only supported with transport security or the GSI Secure Conversation authentication mechanism.
-g, --delegation <mode>	Enables delegation. mode can be either 'limited' or 'full' . Only supported with the GSI Secure Conversation authentication mechanism.
-l, --contextLifetime <value>	Sets the lifetime of the client security context. value is in milliseconds. Only supported with the GSI Secure Conversation authentication mechanism.
-m, --securityMech <type>	Specifies the authentication mechanism. type can be 'msg' for GSI Secure Message, or 'conv' for GSI Secure Conversation.
-c, --serverCertificate <file>	Specifies the server's <i>certificate</i> file used for encryption. Only needed for the GSI Secure Message authentication mechanism.
-p, --protection <type>	Specifies the protection level. type can be 'sig' for signature or 'enc' for encryption.
-x, --proxyFilename <value>	Sets the proxy file to use as client credential.
-z, --authorization <type>	Specifies authorization type. type can be 'self' , 'host' , 'none' , or a string specifying the expected identity of the remote party.
-t, --timeout <timeout>	Specifies client timeout (in seconds). The client will wait maximum of the timeout value for a response from the server before returning an error. By default the timeout value is 10 minutes.

Example:

```
$ ws-enumerate-end -s http://localhost:8080/wsrf/services/ContainerRegistryService \
  enum.context
```

Name

`globus-xpath-query --` Performs XPath query on a resource property document

`globus-xpath-query`

Tool description

The **globus-xpath-query** uses a custom query dialect implementation called TargetedXPath to query the resource property document of a resource. Please see the [querying resource properties using XPath](#) section for more details.

Command syntax

```
globus-xpath-query [options] [query expression] [rpQName]
```

The **query expression** is an XPath expression. The **rpQName** is a resource property QName. If a resource property is specified only that resource property within the resource property document will be queried. Otherwise, the entire resource property document will be queried. By default, a simple XPath query is assumed that returns the entire resource property document.

Table 34. Command-specific options

-n, --nsMapFile <file>	Specifies a file that contains namespace mappings. By default, the <code>etc/globus_wsrf_core/namespace-mappings.xml</code> file is used.
-u, --enumerate	Enumerate the query results. The query response will contain an enumeration context through which the actual query results can be obtained. The returned enumeration context can be used with the ws-enumerate command line tool. Also, please note that by default the enumeration context will expire in 30 minutes.

Table 35. Common options

-h, --help	Displays help information about the command.
-d, --debug	Enables debug mode. For example, full stack traces of errors will be displayed.
-e, --eprFile <file>	Specifies an <i>XML</i> file that contains the <i>WS-Addressing</i> endpoint reference.
-s, --service <url>	Specifies the service URL.
-k, --key <name value>	Specifies the resource key. The name is the QName of the resource key in the string form: {namespaceURI}localPart , while the value is the simple value of the key. For complex keys, use the --eprFile option. Example: <pre>-k "{http://www.globus.org}MyKey" 123</pre>
-f, --descriptor <file>	Specifies a client security descriptor. Overrides all other security settings.
-a, --anonymous	Enables anonymous authentication. Only supported with transport security or the GSI Secure Conversation authentication mechanism.
-g, --delegation <mode>	Enables delegation. mode can be either 'limited' or 'full' . Only supported with the GSI Secure Conversation authentication mechanism.
-l, --contextLifetime <value>	Sets the lifetime of the client security context. value is in milliseconds. Only supported with the GSI Secure Conversation authentication mechanism.
-m, --securityMech <type>	Specifies the authentication mechanism. type can be 'msg' for GSI Secure Message, or 'conv' for GSI Secure Conversation.
-c, --serverCertificate <file>	Specifies the server's <i>certificate</i> file used for encryption. Only needed for the GSI Secure Message authentication mechanism.
-p, --protection <type>	Specifies the protection level. type can be 'sig' for signature or 'enc' for encryption.
-x, --proxyFilename <value>	Sets the proxy file to use as client credential.
-z, --authorization <type>	Specifies authorization type. type can be 'self' , 'host' , 'none' , or a string specifying the expected identity of the remote party.
-t, --timeout <timeout>	Specifies client timeout (in seconds). The client will wait maximum of the timeout value for a response from the server before returning an error. By default the timeout value is 10 minutes.

Examples:

```
$ globus-xpath-query -s http://localhost:8080/wsrfl/services/ContainerRegistryService \
  "/wssg:MemberServiceEPR/wsa:Address"
```

The above command will query the entire resource property document of the service.

```
$ globus-xpath-query -s http://localhost:8080/wsrfl/services/ContainerRegistryService \
  "/wssg:MemberServiceEPR/wsa:Address" wssg:Entry
```

The above command will query only the `wssg:Entry` resource property of the resource property document of the service.

```
$ globus-xpath-query -s http://localhost:8080/wsrf/services/ContainerRegistryService \  
-u "//wssg:MemberServiceEPR/wsa:Address" > enum.context $ ws-enumerate \  
-s http://localhost:8080/wsrf/services/ContainerRegistryService \ -i all enum.context
```

The first command will create an enumeration for the query results and store the returned enumeration context in a file. The second command will use the enumeration context stored in that file to retrieve the actual query results.

DRAFT

Name

Common Java Client Options -- list of common options across commands

Common Java Client Options

Table 36. Common options

-h, --help	Displays help information about the command.
-d, --debug	Enables debug mode. For example, full stack traces of errors will be displayed.
-e, --eprFile <file>	Specifies an <i>XML</i> file that contains the <i>WS-Addressing</i> endpoint reference.
-s, --service <url>	Specifies the service URL.
-k, --key <name value>	Specifies the resource key. The name is the QName of the resource key in the string form: {namespaceURI}localPart , while the value is the simple value of the key. For complex keys, use the --eprFile option. Example: <pre>-k "{http://www.globus.org}MyKey" 123</pre>
-f, --descriptor <file>	Specifies a client security descriptor. Overrides all other security settings.
-a, --anonymous	Enables anonymous authentication. Only supported with transport security or the GSI Secure Conversation authentication mechanism.
-g, --delegation <mode>	Enables delegation. mode can be either ' limited ' or ' full '. Only supported with the GSI Secure Conversation authentication mechanism.
-l, --contextLifetime <value>	Sets the lifetime of the client security context. value is in milliseconds. Only supported with the GSI Secure Conversation authentication mechanism.
-m, --securityMech <type>	Specifies the authentication mechanism. type can be ' msg ' for GSI Secure Message, or ' conv ' for GSI Secure Conversation.
-c, --serverCertificate <file>	Specifies the server's <i>certificate</i> file used for encryption. Only needed for the GSI Secure Message authentication mechanism.
-p, --protection <type>	Specifies the protection level. type can be ' sig ' for signature or ' enc ' for encryption.
-x, --proxyFilename <value>	Sets the proxy file to use as client credential.
-z, --authorization <type>	Specifies authorization type. type can be ' self ', ' host ', ' none ', or a string specifying the expected identity of the remote party.
-t, --timeout <timeout>	Specifies client timeout (in seconds). The client will wait maximum of the timeout value for a response from the server before returning an error. By default the timeout value is 10 minutes.

Chapter 2. Logging

Logging in the Java WS Core is based on the [Jakarta Commons Logging API](http://jakarta.apache.org/commons/logging/)¹.

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

You can find the configuration file at `$GLOBUS_LOCATION/log4j.properties`.

The following example creates a Console Appender called A1, which, by default, logs messages of ERROR or more severe. It is set to log classes under the package `org.globus.wsrfl.impl` under the INFO log level.

```
# Set root category priority to WARN and its only appender to A1.

log4j.rootCategory=ERROR, A1

# A1 is set to be a ConsoleAppender.

log4j.appender.A1=org.apache.log4j.ConsoleAppender

# A1 uses PatternLayout.
log4j.appender.A1.layout=org.apache.log4j.PatternLayout
log4j.appender.A1.layout.ConversionPattern=%d{ISO8601} %-5p %c{2} [%t,%M:%L] %m%n

log4j.category.org.globus.wsrfl.impl=INFO
```

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))

Chapter 3. Troubleshooting

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

1. The standalone container appears to hang

If the standalone container appears to hang, for example the list of deployed services is not shown after a while or all requests to the container time out, please see the [Debugging hanged Java process](#) section for information on how to diagnose this problem.

DRAFT

2. Java WS Core Errors

DRAFT

Table 3.1. Java WS Core Errors

Error Code	Definition
Failed to acquire notification consumer home instance from registry	Caused by <code>javax.naming.NameNotFoundException</code> : Name <code>services</code> is not bound in
The <code>WS-Addressing 'To'</code> request header is missing	This warning is logged by the container if the request did not contain the necessary <code>WS-Addressing</code> headers, those headers at all or is somehow misconfigured.
<code>java.io.IOException: Token length X > 33554432</code>	If you see this error in the container log, it usually means you are trying to connect to HTTPS server using <code>https</code> specifies 8443 as a port number and <code>http</code> as the protocol name.
<code>java.lang.NoSuchFieldError: DOCUMENT</code>	This error usually indicates a mismatch between the version of Apache Axis that the code was compiled with and the version currently running with.
<code>org.globus.wsrfl.InvalidResourceKeyException: Argument key is null / Resource key is missing</code>	These errors usually indicate that a resource key was not passed with the request or that an invalid resource key was used (the element <code>QName</code> of the resource key did not match what the service expected).
Unable to connect to <code>localhost:xxx</code>	Cannot resolve <code>localhost</code> . The machine's <code>/etc/hosts</code> isn't set up correctly and/or you do not have DNS for <code>localhost</code> .
<code>org.globus.common.ChainedIOException: Failed to initialize security context</code>	This may indicate that the user's proxy is invalid.
Error: <code>org.xml.sax.SAXException: Unregistered type: class xxx</code>	This may indicate that an Axis generated XML type, defined by the WS RLS XSD, was not properly registered upon deployment without intervention by the user, sometimes they do not.
No socket factory for <code>'https'</code> protocol	When a client fails with the following exception: <pre>java.io.IOException: No socket factory for 'https' protocol at org.apache.axis.transport.http.HTTPSender.getSocket(HTTPSender.java:100) org.apache.axis.transport.http.HTTPSender.writeToSocket(HTTPSender.java:110) org.apache.axis.transport.http.HTTPSender.invoke(HTTPSender.java:120)</pre> <p>FIXME - it may have happened because...</p>

Error Code	Definition
No client transport named 'https' found	<p>When a client fails with the following exception:</p> <pre>No client transport named 'https' found at org.apache.axis.client.AxisClient.invoke(AxisClient.java:170) at org.apache.axis.client.Call.invokeEngine(Call.java:2726)</pre> <p>The client is most likely loading an incorrect <code>client-config.wsdd</code> configuration file.</p>
ConcurrentModificationException in Tomcat 5.0.x	<p>If the following exception is visible in the Tomcat logs at startup, it might cause the HTTPSValve to fail:</p> <pre>java.util.ConcurrentModificationException at java.util.HashMap\$HashIterator.nextEntry(HashMap.java:782) at java.util.HashMap\$EntryIterator.next(HashMap.java:824) at java.util.HashMap.putAllForCreate(HashMap.java:424) at java.util.HashMap.clone(HashMap.java:656) at mx4j.server.DefaultMBeanRepository.clone(DefaultMBeanRepository.</pre> <p>The HTTPSValve might fail with the following exception:</p> <pre>java.lang.NullPointerException at org.apache.coyote.tomcat5.CoyoteRequest.setAttribute(CoyoteRequestFacade.java:100) at org.apache.coyote.tomcat5.CoyoteRequestFacade.setAttribute(CoyoteRequestFacade.java:100) at org.globus.tomcat.coyote.valves.HTTPSValve.expose(HTTPSVAlve.java:100)</pre> <p>These exceptions will prevent the transport security from working properly in Tomcat.</p>
java.net.SocketException: Invalid argument or cannot assign requested address	<p>FIXME - what causes this?</p>
GAR deploy/undeploy fails with container is running error	<p>A GAR file can only be deployed or undeployed locally while the container is off. However, GAR deployments fail with this error even if the container is off. This usually happens if the container has crashed or was stopped from cleaning up its state files.</p>

3. General troubleshooting information

- In general, if you want to investigate a problem on your own please see [Chapter 10, Debugging](#) for details on how to turn on debugging.
- Most of the command line clients have a `-debug` option that will display more detailed error messages, including the error stack traces.
- [Search the mailing lists](#)¹ such as gt-user@globus.org² or jwscore-user@globus.org³ (before posting a message).
- If you think you have found a bug please report it in our [Bugzilla](#)⁴ system. Please include as much as detail about the problem as possible.

¹ <http://www.globus.org/email-archive-search.php>

² <mailto:gt-user@globus.org>

³ <mailto:jwscore-user@globus.org>

⁴ <http://bugzilla.globus.org/bugzilla/>

Appendix A. Running client programs from command line

Sometimes it might be necessary to run a Java program directly using the `java` executable. There are two recommended ways of doing so (the `GLOBUS_LOCATION` environment variable must first be set in both cases):

Important

If you are not using any of these two methods please take a look at [Appendix B, Running client programs from any directory](#).

1. With `globus-devel-env` script help

The `globus-devel-env` script can be used to set the proper `CLASSPATH` environment variable. To set the `CLASSPATH` on Windows execute:

```
> %GLOBUS_LOCATION%\etc\globus-devel-env.bat
```

On Unix/Linux machines execute (for bash/sh):

```
$ . $GLOBUS_LOCATION/etc/globus-devel-env.sh
```

or (for csh/tcsh):

```
$ source $GLOBUS_LOCATION/etc/globus-devel-env.csh
```

Once the `globus-devel-env` is executed successfully run the Java program, for example:

On Windows:

```
> java -DGLOBUS_LOCATION=%GLOBUS_LOCATION% foo.MyClass
```

On Unix/Linux:

```
$ java -DGLOBUS_LOCATION=$GLOBUS_LOCATION foo.MyClass
```

Note: Passing `-DGLOBUS_LOCATION` is not necessary but will enable the client to execute from any directory.

2. Using bootstrap

Sometimes the above method might fail if the `CLASSPATH` environment variable is too long for the OS to handle. With the bootstrap method, a bootstrap code is executed first which sets the classpath programmatically and then invokes the specified Java program. To invoke a Java program on Windows through bootstrap execute:

```
> java -cp %GLOBUS_LOCATION%\lib\bootstrap.jar  
-DGLOBUS_LOCATION=%GLOBUS_LOCATION% \ org.globus.bootstrap.Bootstrap foo.MyClass
```

On Unix/Linux machines execute:

```
$ java -cp $GLOBUS_LOCATION/lib/bootstrap.jar -DGLOBUS_LOCATION=$GLOBUS_LOCATION \  
org.globus.bootstrap.Bootstrap foo.MyClass
```

Appendix B. Running client programs from any directory

A client launched directly through the `java` executable might fail if ran from a directory other than the `GLOBUS_LOCATION` (this usually happens if the client receives notifications). To ensure that a client can be started from any directory:

- Set the `GLOBUS_LOCATION` system property on the `java` command line.
- Put the `GLOBUS_LOCATION` directory as the very first entry in the classpath used by the client.

For example on Unix/Linux:

```
$ java -DGLOBUS_LOCATION=$GLOBUS_LOCATION -classpath $GLOBUS_LOCATION:mylib.jar  
foo.MyClass
```

or on Windows:

```
> java -DGLOBUS_LOCATION=%GLOBUS_LOCATION% -classpath %GLOBUS_LOCATION%;mylib.jar  
foo.MyClass
```

Appendix C. Usage statistics collection by the Globus Alliance

1. Usage statistics sent by Java WS Core

The following usage statistics are sent by Java WS Core by default in a UDP packet (in addition to the Java WS Core component code, packet version, timestamp, and the source IP address):

- On container startup:
 - container id - random number
 - container type - standalone, servlet, or unknown
 - event type - container startup
 - list of deployed services - service names only
- On container shutdown:
 - container id - random number
 - container type - standalone, servlet, or unknown
 - event type - container shutdown
 - list of activated services - service names only
 - container uptime

If you wish to disable this feature, please see the "Usage Statistics Configuration" section of [Section 2.4, "Usage Statistics Configuration"](#) for instructions.

Also, please see our [policy statement](#)¹ on the collection of usage statistics.

¹ http://www.globus.org/toolkit/docs/4.2/4.2.0/Usage_Stats.html

Glossary

C

certificate A public key plus information about the certificate owner bound together by the digital signature of a CA. In the case of a CA certificate, the certificate is self signed, i.e. it was signed using its own private key.

W

Web Services Addressing (WSA) The WS-Addressing specification defines transport-neutral mechanisms to address web services and messages. Specifically, it defines XML elements to identify web service endpoints and to secure end-to-end endpoint identification in messages. See the [W3C WS Addressing Working Group](#)¹⁴ for details.

X

XML Extensible Markup Language (XML) is standard, flexible, and extensible data format used for web services. See the [W3C XML site](#)²⁰ for details.

¹⁴ <http://www.w3.org/2002/ws/addr/>

²⁰ <http://www.w3.org/XML/>

Index

C

- client programs
 - running client programs
 - from any directory, 63
 - from command line, 62
- configuring
 - globus-update-client-config, 39
 - Merges client-config.wsdd files from deployed modules into the global client-config.wsdd configuration file, 39
 - validating configuration files of all services, 40
- containers
 - globus-reload-container, 41
 - globus-start-container, 4
 - globus-start-container-detached, 7
 - globus-stop-container, 5
 - globus-stop-container-detached, 8
 - reinitializing the standalone container, 41
 - starting standalone container, 4
 - starting standalone container detached from controlling TTY, 7
 - stopping standalone container, 5
 - stopping standalone container detached from controlling TTY, 8

E

- enumeration
 - retrieving enumeration data, 49
 - starting an enumeration, 47
 - stopping an enumeration, 51
 - ws-enumerate, 49
 - ws-enumerate-end, 51
 - ws-enumerate-start, 47
- environment
 - displaying component version information and validating JVM version, 37
 - displaying remote component version information, 38
 - globus-check-environment, 37
 - globus-check-remote-environment, 38
- errors, 59

G

- globus-validate-descriptors, 40
- Grid Archive (GAR)
 - deploying a GAR file
 - locally, 33
 - remotely, 45
 - globus-deploy-gar, 33
 - globus-remote-deploy-gar, 45

- globus-remote-undeploy-gar, 43
- globus-undeploy-gar, 35
- undeploying a GAR file
 - locally, 35
 - remotely, 43

R

- resource
 - destroying a resource, 9
 - setting the termination time of a resource, 11
 - wsrf-destroy, 9
 - wsrf-set-termination-time, 11
- resource properties
 - deleting a resource property, 23
 - getting a single resource property from a resource, 15
 - getting multiple resource properties from a resource, 17
 - globus-xpath-query, 53
 - inserting a property into the resource property document of a resource, 19
 - performing an XPath query on a resource property document, 53
 - querying the resource property document of a resource, 13
 - updating the property value in the resource property document of a resource, 21
 - wsrf-delete-property, 23
 - wsrf-get-properties, 17
 - wsrf-get-property, 15
 - wsrf-insert-property, 19
 - wsrf-query, 13
 - wsrf-update-property, 21
- running client programs
 - from any directory, 63
 - from command line, 62

S

- subscription
 - getting a current message associated with a topic, 25
 - pausing a subscription, 27
 - resuming a subscription, 29
 - subscribing to a topic, 31
 - wsn-get-current-message, 25
 - wsn-pause-subscription, 27
 - wsn-resume-subscription, 29
 - wsn-subscribe, 31

T

- troubleshooting
 - for end-users, 58

U

usage statistics, 64

V

validating configuration files of all services, 40

X

XPath

globus-xpath-query, 53

performing an XPath query on a resource property document, 53

DRAFT