

GT 4.0 WS MDS WebMDS

GT 4.0 WS MDS WebMDS

Table of Contents

Docs that relate to all MDS components: Key Concepts, Migrating Guide and Samples	vi
1. 4.0.0 Release Notes	1
1. Component Overview	1
2. Feature Summary	1
3. Bug Fixes	1
4. Known Problems	2
5. Technology Dependencies	2
6. Tested Platforms	2
7. Backward Compatibility Summary	3
8. For More Information	4
2. 4.0.1 Release Notes	5
1. Introduction	5
2. Changes Summary	5
3. Bug Fixes	5
4. Known Problems	5
5. For More Information	5
3. 4.0.2 Release Notes	6
1. Introduction	6
2. Changes Summary	6
3. Bug Fixes	6
4. Known Problems	6
5. For More Information	6
4. 4.0.3 Release Notes	7
1. Introduction	7
2. Changes Summary	7
3. Bug Fixes	7
4. Known Problems	7
5. For More Information	7
5. 4.0.4 Release Notes	8
1. Introduction	8
2. Changes Summary	8
3. Bug Fixes	8
4. Known Problems	8
5. For More Information	8
6. System Administrator's Guide	9
1. Introduction	9
2. Building and Installing	9
3. Configuring	9
4. Deploying	12
5. Testing	14
6. Security Considerations	17
7. Troubleshooting	17
7. User's Guide	19
1. Introduction	19
2. Command-line tools	19
3. Graphical user interfaces	19
4. Troubleshooting	20
8. Developer's Guide	21
1. Introduction	21
2. Before you begin	21
3. Architecture and design overview	23

4. Public interface	24
5. Usage scenarios	24
6. Troubleshooting	24
7. Related Documentation	24
9. Fact Sheet	25
1. Brief component overview	25
2. Summary of features	25
3. Usability summary	25
4. Backward compatibility summary	25
5. Technology dependencies	26
6. Tested platforms	26
7. Associated standards	27
8. For More Information	27
10. Guide to Public Interfaces	28
1. Semantics and syntax of APIs	28
2. Semantics and syntax of the WSDL	29
3. Command-line tools	29
4. Overview of Graphical User Interface	30
5. Semantics and syntax of domain-specific interface	31
6. Configuration interface	31
7. Environment variable interface	34
11. Quality Profile	35
1. Test coverage reports	35
2. Code analysis reports	35
3. Outstanding bugs	35
4. Bug Fixes	35
5. Performance reports	36
GT 4.0 WS MDS Glossary	37

List of Tables

6.1. Pre-configured information sources	10
6.2. Configuration parameters used with FileXMLSource	11
6.3. Configuration parameters used with NodeXMLSource	11
6.4. Configuration parameters used with ResourcePropertyQueryNodeSource	11
6.5. Configuration parameters used with ResourcePropertyNodeSource	12
7.1. Form arguments used by WebMDS	20
10.1. Form arguments used by WebMDS	30
10.2. Pre-configured information sources	32
10.3. Configuration parameters used with FileXMLSource	33
10.4. Configuration parameters used with NodeXMLSource	33
10.5. Configuration parameters used with ResourcePropertyQueryNodeSource	33
10.6. Configuration parameters used with ResourcePropertyNodeSource	34

Docs that relate to all MDS components: Key Concepts, Migrating Guide and Samples

- [Key Concepts](#)¹
- [Migrating Guide](#)²
- [Samples](#)³

¹ ../key-index.html

² ../WS_MDS_Migrating_Guide.html

³ ../WS_MDS_Samples.html

Chapter 1. GT 4.0 Development Release Notes for WS MDS WebMDS

1. Component Overview

WebMDS enables end users to view monitoring information via a standard web browser interface, without installing any additional software on their PC. WebMDS is implemented as a servlet that uses a plugin interface to gather monitoring information (or any other information in XML format) and XSLT transforms, and present the data to the user in a readable form. Web site administrators can customize their own WebMDS deployments by using HTML form options, configuring different plugins to collect data and XSLT transforms, and creating their own plugins and XSLT transforms.

2. Feature Summary

Features new in release 4.0:

- Extensible plugin interface to support various mechanisms to gather monitoring information and XSLT transforms.
- Plugins to acquire monitoring information via resource property mechanisms.
- Plugin to acquire XSLT transforms by reading from local files.

Other Supported Features

- WebMDS is a new component, so all its features are "new in release 4.0".

Deprecated Features

- None

3. Bug Fixes

- [2347: Information missing from service group entry detail page in some cases](http://bugzilla.globus.org/globus/show_bug.cgi?id=2347)¹
- [2275: Misleading summary information in servicegrouptable.xsl](http://bugzilla.globus.org/globus/show_bug.cgi?id=2275)²
- [2257: Add RFT support into servicegroup table XSL](http://bugzilla.globus.org/globus/show_bug.cgi?id=2257)³
- [2143: stylesheet that displays simple table](http://bugzilla.globus.org/globus/show_bug.cgi?id=2143)⁴
- [2806: Fix default values in "open-ended resource property query" sample form](http://bugzilla.globus.org/globus/show_bug.cgi?id=2806)⁵
- [2769: webmds has missing files in filelists](http://bugzilla.globus.org/globus/show_bug.cgi?id=2769)⁶

¹ http://bugzilla.globus.org/globus/show_bug.cgi?id=2347

² http://bugzilla.globus.org/globus/show_bug.cgi?id=2275

³ http://bugzilla.globus.org/globus/show_bug.cgi?id=2257

⁴ http://bugzilla.globus.org/globus/show_bug.cgi?id=2143

⁵ http://bugzilla.globus.org/globus/show_bug.cgi?id=2806

⁶ http://bugzilla.globus.org/globus/show_bug.cgi?id=2769

- [All fixed bugs and enhancement requests for WebMDS](#)⁷

4. Known Problems

- [3040: WebMDS can break if started from wrong directory](#)⁸
- [3051: Handle huge indexes](#)⁹
- [3160: Format summary line for RLS](#)¹⁰
- [All open bug reports and enhancement requests for WebMDS](#)¹¹

5. Technology Dependencies

WebMDS depends on the following GT components:

- Java WS Core

WebMDS depends on the following 3rd party software:

- [Tomcat](#)¹²

6. Tested Platforms

Tested Platforms for WebMDS:

- The WebMDS server has only been tested with Tomcat version 5.0.28; it has been tested on RedHat Linux (i386) and, to a lesser extent, on Windows XP.
- On the client side, WebMDS should be accessible from any web browser on any platform.

6.1. Installing WebMDS on Windows

Although the WebMDS server is not officially supported on non-Unix platforms, and no Windows installer exists for WebMDS, it is possible to run WebMDS on Windows. The following instructions describe how to install WebMDS on a Windows platform.

1. Install [Tomcat](#)¹³ and set your CATALINA_HOME environment variable to the directory into which Tomcat was installed.

⁷ http://bugzilla.globus.org/globus/buglist.cgi?short_desc_type=allwordssubstr&short_desc=&product=MDS&component=wsrf_webmds&long_desc_type=allwordssubstr&long_desc=&bug_file_loc_type=allwordssubstr&bug_file_loc=&bug_status=RESOLVED&bug_status=VERIFIED&bug_status=CLOSED&emailtype1=substring&email1=&emailtype2=substring&email2=&bugidtype=include&bug_id=&votes=&changedin=&chfieldfrom=&chfieldto=Now&chfieldvalue=&cmdtype=doit&newqueryname=&order=Reuse+same+sort+as+last+time&field0-0-0=noop&type0-0-0=noop&value0-0-0=

⁸ http://bugzilla.globus.org/globus/show_bug.cgi?id=3040

⁹ http://bugzilla.globus.org/globus/show_bug.cgi?id=3051

¹⁰ http://bugzilla.globus.org/globus/show_bug.cgi?id=3160

¹¹ http://bugzilla.globus.org/globus/buglist.cgi?short_desc_type=allwordssubstr&short_desc=&product=MDS&component=wsrf_webmds&long_desc_type=allwordssubstr&long_desc=&bug_file_loc_type=allwordssubstr&bug_file_loc=&bug_status=NEW&bug_status=AS-SIGNED&bug_status=REOPENED&emailtype1=substring&email1=&emailtype2=substring&email2=&bugidtype=include&bug_id=&votes=&changedin=&chfieldfrom=&chfieldto=Now&chfieldvalue=&cmdtype=doit&newqueryname=&order=Reuse+same+sort+as+last+time&field0-0-0=noop&type0-0-0=noop&value0-0-0=

¹² <http://jakarta.apache.org/tomcat/>

¹³ <http://jakarta.apache.org/tomcat/>

2. Install the Globus Java WS-Core distribution from the [Globus Toolkit download page](#)¹⁴. Set your GLOBUS_LOCATION environment variable to the directory into which you installed Globus Java WS-Core
3. Check the ws-mds distribution out of the [Globus CVS repository](#)¹⁵, using the globus_4_0_branch tag.
4. Install the servicegroup package:

```
cd c:\wherever\ws-mds\servicegroup\schema
ant deploy
cd ..\source
ant deploy
```

where *wherever* is the directory into which you checked out the ws-mds sources.

5. Install WebMDS:

```
cd c:\wherever\ws-mds\webmds
ant deploy
```

6. Create the webmds context file (this tells Tomcat where to find WebMDS):

```
%GLOBUS_LOCATION%\lib\webmds\bin\webmds-create-context-file %CATALINA_HOME%\conf\Catali
```

7. Restart Tomcat.

WebMDS can then be configured and used as described in the rest of the [WebMDS documentation](#)¹⁶.

7. Backward Compatibility Summary

Protocol changes since GT version 3.2:

- WebMDS did not exist in GT version 3.2.

API changes since GT version 3.2:

- WebMDS did not exist in GT version 3.2.

Exception changes since GT version 3.2:

- WebMDS did not exist in GT version 3.2.

Schema changes since GT version 3.2:

- WebMDS did not exist in GT version 3.2.

¹⁴ <http://www.globus.org/toolkit/downloads/>

¹⁵ <http://www.globus.org/toolkit/docs/development/remote-cvs.html>

¹⁶ [index.html](#)

8. For More Information

Click [here](#)¹⁷ for more information about this component.

¹⁷ [index.html](#)

Chapter 2. GT 4.0.1 Incremental Release Notes: WS MDS WebMDS

1. Introduction

These release notes are for the incremental release 4.0.1. It includes a summary of changes since 4.0.0, bug fixes since 4.0.0 and any known problems that still exist at the time of the 4.0.1 release. This page is in addition to the top-level 4.0.1 release notes at <http://www.globus.org/toolkit/releasenotes/4.0.1>.

For release notes about 4.0 (including feature summary, technology dependencies, etc) go to the [WS MDS WebMDS 4.0 Release Notes](#)¹.

2. Changes Summary

Other than bug fixes, no changes have occurred for WebMDS.

3. Bug Fixes

The following bugs were fixed for WebMDS:

- [Bug 3160](#):² Format summary line for RLS
- [Bug 3537](#):³ WebMDS security -- remote users can determine some information about local files
- [Bug 3617](#):⁴ Some xsl files missing from webmds filelist

4. Known Problems

The following problems are known to exist for WebMDS at the time of the 4.0.1 release:

- [Bug 3040](#):⁵ WebMDS can break if started from wrong directory
- [Bug 3051](#):⁶ Handle huge indexes

5. For More Information

Click [here](#)⁷ for more information about this component.

¹ http://www.globus.org/toolkit/docs/4.0/info/webmds/WS_MDS_WebMDS_Release_Notes.html

² http://bugzilla.globus.org/globus/show_bug.cgi?id=3160

³ http://bugzilla.globus.org/globus/show_bug.cgi?id=3537

⁴ http://bugzilla.globus.org/globus/show_bug.cgi?id=3617

⁵ http://bugzilla.globus.org/globus/show_bug.cgi?id=3040

⁶ http://bugzilla.globus.org/globus/show_bug.cgi?id=3051

⁷ [index.html](#)

Chapter 3. GT 4.0.2 Incremental Release Notes: WS MDS WebMDS

1. Introduction

These release notes are for the incremental release 4.0.2. It includes a summary of changes since 4.0.1, bug fixes since 4.0.1 and any known problems that still exist at the time of the 4.0.2 release. This page is in addition to the top-level 4.0.2 release notes at <http://www.globus.org/toolkit/releasenotes/4.0.2>.

For release notes about 4.0 (including feature summary, technology dependencies, etc) go to the [WS MDS WebMDS 4.0 Release Notes](#)¹.

2. Changes Summary

Since 4.0.1, some new example XSL transform files have been added to the WebMDS distribution.

3. Bug Fixes

No bugs have been fixed for WebMDS since 4.0.1.

4. Known Problems

The following problems are known to exist for WebMDS at the time of the 4.0.2 release:

- [Bug 3040](#):² WebMDS can break if started from wrong directory
- [Bug 3051](#):³ Handle huge indexes

5. For More Information

Click [here](#)⁴ for more information about this component.

¹ http://www.globus.org/toolkit/docs/4.0/info/webmds/WS_MDS_WebMDS_Release_Notes.html

² http://bugzilla.globus.org/globus/show_bug.cgi?id=3040

³ http://bugzilla.globus.org/globus/show_bug.cgi?id=3051

⁴ [index.html](#)

Chapter 4. GT 4.0.3 Incremental Release Notes: WS MDS WebMDS

1. Introduction

These release notes are for the incremental release 4.0.3. It includes a summary of changes since 4.0.2, bug fixes since 4.0.2 and any known problems that still exist at the time of the 4.0.3 release. This page is in addition to the top-level 4.0.3 release notes at <http://www.globus.org/toolkit/releasenotes/4.0.3>.

For release notes about 4.0 (including feature summary, technology dependencies, etc) go to the [WS MDS WebMDS 4.0 Release Notes](#)¹.

2. Changes Summary

No changes have been made to WebMDS since GT 4.0.2.

3. Bug Fixes

No bugs have been fixed for WebMDS since 4.0.2.

4. Known Problems

The following problems are known to exist for WebMDS at the time of the 4.0.3 release:

- [Bug 3040](#):² WebMDS can break if started from wrong directory
- [Bug 3051](#):³ Handle huge indexes

5. For More Information

Click [here](#)⁴ for more information about this component.

¹ http://www.globus.org/toolkit/docs/4.0/info/webmds/WS_MDS_WebMDS_Release_Notes.html

² http://bugzilla.globus.org/globus/show_bug.cgi?id=3040

³ http://bugzilla.globus.org/globus/show_bug.cgi?id=3051

⁴ [index.html](#)

Chapter 5. GT 4.0.4 Incremental Release Notes: WS MDS WebMDS

1. Introduction

These release notes are for the incremental release 4.0.4. It includes a summary of changes since 4.0.3, bug fixes since 4.0.3 and any known problems that still exist at the time of the 4.0.4 release. This page is in addition to the top-level 4.0.4 release notes at <http://www.globus.org/toolkit/releasenotes/4.0.4>.

For release notes about 4.0 (including feature summary, technology dependencies, etc) go to the [WS MDS WebMDS 4.0 Release Notes](#)¹.

2. Changes Summary

No changes have been made to WebMDS since GT 4.0.3.

3. Bug Fixes

No bugs have been fixed for WebMDS since 4.0.3.

4. Known Problems

- [Bug 3040](#):² WebMDS can break if started from wrong directory
- [Bug 3051](#):³ Handle huge indexes

5. For More Information

Click [here](#)⁴ for more information about this component.

¹ http://www.globus.org/toolkit/docs/4.0/info/webmds/WS_MDS_WebMDS_Release_Notes.html

² http://bugzilla.globus.org/globus/show_bug.cgi?id=3040

³ http://bugzilla.globus.org/globus/show_bug.cgi?id=3051

⁴ [index.html](#)

Chapter 6. GT 4.0 WS MDS WebMDS: System Administrator's Guide

1. Introduction

WebMDS enables end users to view monitoring information via a standard web browser interface, without installing any additional software on their PC. WebMDS is implemented as a servlet that uses a plugin interface to gather monitoring information (or any other information in XML format) and XSLT transforms, and present the data to the user in a readable form. Web site administrators can customize their own WebMDS deployments by using HTML form options, configuring different plugins to collect data and XSLT transforms, and creating their own plugins and XSLT transforms.

This guide contains advanced configuration information for system administrators working with WS MDS WebMDS. It provides references to information on procedures typically performed by system administrators, including installation, configuring, deploying, and testing the installation.

This information is in addition to the basic Globus Toolkit prerequisite, overview, installation, security configuration instructions in the [GT 4.0 System Administrator's Guide](#)¹. Read through this guide before continuing!

2. Building and Installing

WebMDS is built and installed as part of the standard Globus Toolkit installation.

3. Configuring

3.1. Configuration overview

WebMDS can be configured to get information from any of various sources and to filter it through any XSL transform. WebMDS uses configuration files to specify the location of (and to name) sources of information and xsl and web form arguments to select among these configured information sources and xsl transforms.

By default, WebMDS comes configured to report information about an index server using transaction-level security on the default port (8443) on the local system. If you are running the Globus Toolkit in this default configuration, then you can use WebMDS to query your local *Index Service* without any configuration changes.

If you wish to monitor a different index server, you will need to edit the file `$GLOBUS_LOCATION/lib/webm-ds/conf/indexinfo` to change the URL in the line:

```
<value>https://127.0.0.1:8443/wsrf/services/DefaultIndexService</value>
```

to match the URL of your default index service. Changes to WebMDS configuration files take effect the next time that Tomcat is restarted.

For other configuration changes (e.g., monitoring different kinds of services), see the detailed configuration information below.

¹ <http://www.globus.org/toolkit/docs/4.0/admin/docbook/>

3.2. Syntax of the interface

Each configuration file in `$GLOBUS_LOCATION/lib/webmds/conf` defines a source of XML, which can be used in an HTML form to specify sources of information and XSL transforms. The distribution contains some standard configuration files in this directory, including:

Table 6.1. Pre-configured information sources

<code>indexinfo</code>	all resource properties from an index server running with transaction-level security on port 8443 on the local host
<code>indexinfo_nosec</code>	all resource properties from an index server running with no security on port 8080 on the local host
<code>openEndedQuery</code>	all resource properties from a user-specified grid service
<code>openEndedRP</code>	a user-specified resource property from a user-specified grid service
<code>servicegroupxml</code>	an xsl transform that presents summary information about a service group
<code>sgedetail</code>	an xsl transform that presents detailed information about a service group entry

Each configuration file defines a `WebmdsConfig` object. A `WebmdsConfig` object consists of:

- A **description**: a textual description of the XML source being defined.
- A **className**: the name of the Java class that will be used to acquire the XML data.
- Zero or more **parameter** objects, each of which consists of the **name** of some parameter recognized by the Java class specified by **className**, and the **string value** of that parameter.

For example, this is `$GLOBUS_LOCATION/lib/webmds/conf/servicegroupxml`, which defines the `servicegroupxml` XML source:

```
<WebmdsConfig>
  <description>
    XSL file to show service group summary information
  </description>
  <className>org.globus.mds.webmds.xmlSources.file.FileXmlSource</className>
  <parameter>
    <name>file</name>
    <value>xslfiles/servicegrouptable.xsl</value>
  </parameter>
</WebmdsConfig>
```

This file tells WebMDS to use the `org.globus.mds.webmds.xmlSources.file.FileXmlSource` Java class (a class which reads XML from a local file) to collect XML data and to pass a `file` parameter (which that Java class interprets as the name of the file to open, relative to the WebMDS base directory).

Tomcat must be restarted (or one of the more advanced Tomcat administrative mechanisms must be used) for changes to these configuration files to take effect.

3.3. XML Sources included with WebMDS

3.3.1. FileXMLSource

The class `org.globus.mds.webmds.xmlSources.file.FileXmlSource` reads XML from a file, and recognizes a single parameter:

Table 6.2. Configuration parameters used with FileXMLSource

<code>file</code>	The name of the file to read. Relative path names are interpreted relative to the WebMDS base directory (<code>\$GLOBUS_LOCATION/lib/webmds</code>).
-------------------	--

3.3.2. NodeXMLSource

This XML source class uses a `WebmdsNodeSource` object to fetch an XML document and return it in a form that is usable by WebMDS. It recognizes the following options:

Table 6.3. Configuration parameters used with NodeXMLSource

<code>class</code>	The name of a class that implements the <code>WebmdsNodeSource</code> interface. An instance of this class will be used to get an XML document.
<code>parameters</code>	Additional parameters are passed to an instance of the class specified by the <code>class</code> argument.

3.3.3. Classes That Implement WebmdsNodeSource

The following classes implement the `NodeXMLSource` interfaces and can be used in conjunction with `NodeXMLSource`

3.3.4. ResourcePropertyQueryNodeSource

This class performs a resource property query to get all the resource properties for some web service. It recognizes the following configuration parameters:

Table 6.4. Configuration parameters used with ResourcePropertyQueryNodeSource

<code>endpoint</code>	The endpoint name to be used in a resource property query.
<code>endpointKeyName</code> and <code>endpointKeyValue</code>	An optional key/value pair to use as reference properties for the endpoint specified with the <code>endpoint</code> parameter.
<code>allowUserEndpoints</code>	If <code>true</code> , values for <code>xmlSource.sourceName.param.endpoint</code> , <code>xmlSource.sourceName.param.endpointKeyName</code> , and <code>xmlSource.sourceName.param.endpointKeyValue</code> specified in the request will override the configured endpoint value.
<code>endpointFile</code>	The name of a file from which the endpoint information (in XML) will be read. This configuration parameter can never be overridden by request arguments.

3.3.5. ResourcePropertyNodeSource

This class queries a web service for a single resource property. It recognizes the following parameters:

Table 6.5. Configuration parameters used with ResourcePropertyNodeSource

endpoint	The endpoint name to be used in a resource property query.
endpointKeyName and endpointKeyValue	An optional key/value pair to use as reference properties for the endpoint specified with the endpoint parameter.
allowUserEndpoints	If true, values for <code>xmlSource.sourceName.param.endpoint</code> , <code>xmlSource.sourceName.param.endpointKeyName</code> , and <code>xmlSource.sourceName.param.endpointKeyValue</code> specified in the request will override the configured endpoint value.
endpointFile	The name of a file from which the endpoint information (in XML) will be read. This configuration parameter can never be overridden by request arguments.
rpNamespace	The namespace part of the QName of the resource property to be queried for.
rpName	The local name part of the QName of the resource property to be queried for.
allowUserResourceProperties	If true, values of <code>xmlSource.sourceName.param.rpNamespace</code> and <code>xmlSource.sourceName.param.rpNames</code> specified in the request will override the configured resource property namespace and name.

4. Deploying

Because WebMDS is implemented as a servlet, it must be deployed into a servlet container, such as [Tomcat](http://jakarta.apache.org/tomcat/)². The following instructions assume that you've installed Tomcat version 5.0.28, and set the `$CATALINA_HOME` environment variable to the directory into which you've installed Tomcat.

4.1. Standard deployment into Tomcat 5.0.28

The standard deployment consists of two steps: creating a configuration file that tells Tomcat where to find the WebMDS servlet and related files, and restarting Tomcat so that it will read this new configuration file. These steps require write permission on files and directories in `$CATALINA_HOME`; they do not require write permission on anything in `$GLOBUS_LOCATION`.

To create the configuration file, run this command:

```
$GLOBUS_LOCATION/lib/webmds/bin/webmds-create-context-file \  
$CATALINA_HOME/conf/Catalina/localhost
```

This will create `$CATALINA_HOME/conf/Catalina/localhost/webmds.xml`. Note: if this file already exists (e.g., if you've previously installed another version of WebMDS), you'll need to use the `-f` option to `webmds-create-context-file`.

Next, make sure that Tomcat has a version of the Xalan library (used by WebMDS to do XSL transforms) that is compatible with the one used by Globus:

```
cp $GLOBUS_LOCATION/endorsed/xalan.jar $CATALINA_HOME/common/endorsed/.
```

Next, restart Tomcat. If Tomcat is already running, stop it:

² <http://jakarta.apache.org/tomcat/>

```
$CATALINA_HOME/bin/shutdown.sh
```

Then, start Tomcat:

```
$CATALINA_HOME/bin/startup.sh
```

4.2. Deploying WebMDS and Globus in the same Tomcat Server

If you wish to run Globus and WebMDS in the same Tomcat instance (instead of, for example, running Globus in the Globus standalone container and WebMDS in Tomcat), then do the following:

1. Install Globus and deploy it into Tomcat, as described in the [GT4 Admin Guide](#)³.
2. Run **webmds-create-context-file**:

```
$GLOBUS_LOCATION/lib/webmds/bin/webmds-create-context-file \  
$CATALINA_HOME/conf/Catalina/localhost
```

(see the previous section for more details about **webmds-create-context-file**).

3. The Globus and WebMDS deployments install identical copies of certain files in different places. The presence of these duplicates causes WebMDS to fail when sending requests to secure servers. To prevent this problem, remove the duplicates:

```
rm $GLOBUS_LOCATION/lib/webmds/WEB-INF/lib/puretls.jar  
rm $GLOBUS_LOCATION/lib/webmds/WEB-INF/lib/cryptix*.jar  
rm $GLOBUS_LOCATION/lib/webmds/WEB-INF/lib/jce-jdk*.jar
```

4. Finally, restart Tomcat. If Tomcat is already running, stop it:

```
$CATALINA_HOME/bin/shutdown.sh
```

Then, start Tomcat:

```
$CATALINA_HOME/bin/startup.sh
```

4.3. Custom deployment

If you are already running a Tomcat server (or other server that supports servlets) and your preferred mechanism for installing servlets is something other than creating a configuration file and restarting your web server, feel free to use that mechanism. The servlet root for WebMDS is `$GLOBUS_LOCATION/lib/webmds`.

³ <http://www.globus.org/toolkit/docs/4.0/admin/docbook/>

For the rest of these instructions, the term *Globus user* will be used to refer to the owner of the `$GLOBUS_LOCATION` directory, and *Tomcat user* will be used to refer to the owner of the `$CATALINA_HOME` directory. If the Globus and Tomcat installations were performed from the same user account, the Globus user and Tomcat user will be the same.

Any time you change the servlet configuration (or any jar files used by the servlet), you'll need to let tomcat know there was a change. If you have a preferred way of configuring tomcat, feel free to use it, with `$GLOBUS_LOCATION/lib/webmds` as the servlet directory. These steps need to be performed by the Tomcat user.

If you're using tomcat 5.0.28 and haven't done any custom configuration (such as defining additional hosts) other than changing the tomcat port, you can configure tomcat by doing the following:

1. Create a context descriptor file called `webmds.xml` in the location where tomcat will look for it:

```
$GLOBUS_LOCATION/lib/webmds/bin/webmds-create-context-file \  
$CATALINA_HOME/conf/Catalina/localhost
```



Note

If the file `$CATALINA_HOME/conf/Catalina/localhost/webmds.xml` already exists, you can use the `-f` flag to `create-context-file` to overwrite it. to the tomcat configuration directory.

2. If tomcat is running, shut it down.

```
$CATALINA_HOME/bin/shutdown.sh
```

3. Start tomcat up.

```
$CATALINA_HOME/bin/startup.sh
```

5. Testing

The easiest way to test your installation is to use it to view your *Index Service*, by pointing your web browser at `http://your-tomcat-host:your-tomcat-port/webmds` and clicking on the link labelled "A list of resources registered to the local default index service".

For more in-depth tests, you can run the WebMDS unit tests, by doing the following:

1. Install [httpunit](http://httpunit.sourceforge.net)⁴, version 1.6 or later. Set the environment variable `GLOBUS_HTTPUNIT_DIR` to the directory into which httpunit has been installed.
2. Install the WebMDS test package; from the GT4 distribution directory, run

```
make gt4-webmds-test
```

3. Run the core WebMDS test suite. This tests the WebMDS servlet itself, the File XML Source, and the more commonly-used xslt transforms. There are two modes in which this test suite can be run.

⁴ <http://httpunit.sourceforge.net>

- The core WebMDS tests can be run in a servlet container simulator. This tests the WebMDS code but does not test whether or not WebMDS has been deployed correctly into Tomcat:

```
ant -f $GLOBUS_LOCATION/etc/globus_wsrf_mds_webmds_test/build.xml test-installed
```

The output should look something like this:

```
Buildfile: GLOBUS_LOCATION/etc/globus_wsrf_mds_webmds_test/build.xml
```

```
test-installed:
```

```
[junit] Running org.globus.mds.webmds.test.PackageTests
[junit] Running org.globus.mds.webmds.test.SimpleServletTest tests with servlet
[junit] No webmds.test.servletURL property specified; skipping org.globus.mds.web
[junit] Running org.globus.mds.webmds.test.ServletXslTests tests with servlet si
[junit] No webmds.test.servletURL property specified; skipping org.globus.mds.web
[junit] Tests run: 8, Failures: 0, Errors: 0, Time elapsed: 4.516 sec
```

```
BUILD SUCCESSFUL
```

```
Total time: 8 seconds
```

- The core WebMDS tests can be run against a running WebMDS server, to test the local WebMDS deployment:

```
ant \
  -f $GLOBUS_LOCATION/etc/globus_wsrf_mds_webmds_test/build.xml \
  "-Dwebmds.test.servletURL=http://webmds_host:webmds_port/webmds/webmds" \
  test-installed
```

The output should look something like this:

```
Buildfile: GLOBUS_LOCATION/etc/globus_wsrf_mds_webmds_test/build.xml
```

```
test-installed:
```

```
[junit] Running org.globus.mds.webmds.test.PackageTests
[junit] Running org.globus.mds.webmds.test.SimpleServletTest tests with servlet
[junit] Running org.globus.mds.webmds.test.SimpleServletTest tests against serve
[junit] Running org.globus.mds.webmds.test.ServletXslTests tests with servlet si
[junit] Running org.globus.mds.webmds.test.ServletXslTests tests against server
[junit] Tests run: 8, Failures: 0, Errors: 0, Time elapsed: 5.229 sec
```

```
BUILD SUCCESSFUL
```

```
Total time: 8 seconds
```

The tests have passed if the number of failures and number of errors are both 0. Detailed test output can be found in the file `$GLOBUS_LOCATION/etc/globus_wsrf_mds_webmds_test/test-reports/TEST-org.globus.mds.webmds.test.PackageTests.xml`.

4. Run the WebMDS resource property node source test suite, to test the ability of WebMDS to query a running MDS4 Index Server. This test suite requires that both a secure Index server and an insecure Index server be running. As with the core tests, the resource property tests may be run in two modes.

- The tests can be run in a servlet container simulator. This tests the WebMDS code, and the interaction between the WebMDS code and running Index servers, but does not test whether or not WebMDS has been deployed correctly into tomcat:

```
ant -f \
  $GLOBUS_LOCATION/etc/globus_wsrf_mds_webmds_resource_property_source_test/build.xml
  "-Dwebmds.rpTest.insecureServicePrefix=http://index_server_host:index_server_port/"
  "-Dwebmds.rpTest.secureServicePrefix=https://index_server_host:index_server_port/"
  test-installed
```

The output should look something like this:

Buildfile: *GLOBUS_LOCATION*/etc/globus_wsrf_mds_webmds_resource_property_source_test/

```
test-installed:
[junit] Running org.globus.mds.webmds.xmlSources.resourceProperties.test.Package
[junit] querying resource properties at 'http://insecure_index_server_host:insec
[junit] querying resource properties at 'https://secure_index_server_host:secure
[junit] Tests will use Globus servers at https://secure_index_server_host:secure
[junit] Running org.globus.mds.webmds.xmlSources.resourceProperties.test.Resource
[junit] Tests will use Globus servers at https://secure_index_server_host:secure
[junit] No webmds.test.servletURL property specified; skipping org.globus.mds.we
[junit] Tests will use Globus servers at https://secure_index_server_host:secure
[junit] Tests run: 5, Failures: 0, Errors: 0, Time elapsed: 6.626 sec
```

BUILD SUCCESSFUL

Total time: 10 seconds

- To run an end-to-end test that tests the communication between a deployed WebMDS server and running index servers, do the following:

```
ant -f \
  $GLOBUS_LOCATION/etc/globus_wsrf_mds_webmds_resource_property_source_test/build.xml
  "-Dwebmds.rpTest.insecureServicePrefix=http://insecure_index_server_host:index_ser
  "-Dwebmds.rpTest.secureServicePrefix=https://secure_index_server_host:index_server
  "-Dwebmds.test.servletURL=http://webmds_host:webmds_port/webmds/webmds" \
  test-installed
```

The output should look something like this:

Buildfile: *GLOBUS_LOCATION*/etc/globus_wsrf_mds_webmds_resource_property_source_test/

```
test-installed:
[junit] Running org.globus.mds.webmds.xmlSources.resourceProperties.test.Package
[junit] querying resource properties at 'http://insecure_index_server_host:insec
[junit] querying resource properties at 'https://secure_index_server_host:secure
[junit] Tests will use Globus servers at https://secure_index_server_host:secure
[junit] Running org.globus.mds.webmds.xmlSources.resourceProperties.test.Resource
[junit] Tests will use Globus servers at https://secure_index_server_host:secure
[junit] Running org.globus.mds.webmds.xmlSources.resourceProperties.test.Resource
```

```
[junit] Tests will use Globus servers at https://secure_index_server_host:secure
[junit] Tests run: 5, Failures: 0, Errors: 0, Time elapsed: 7.041 sec
```

```
BUILD SUCCESSFUL
Total time: 10 seconds
```

The tests have passed if the number of failures and number of errors are both 0. Detailed test output can be found in the file `$GLOBUS_LOCATION/etc/globus_wsrp_mds_webmds_resource_property_source_test/test-reports/TEST-org.globus.mds.webmds.xmlSources.resource-Properties.test.PackageTests.xml`.

6. Security Considerations

By default, the WebMDS plugins distributed as part of the Toolkit do not use authentication credentials -- they retrieve information using anonymous SSL authentication or no authentication at all, and thus retrieve only publicly-available information.

The `ResourcePropertyNodeSource` and `ResourcePropertyQueryNodeSource` plugins can be configured either to allow users to specify what resources they want to query or to only allow users to query resources pre-configured by the web administrator. The standard WebMDS deployment allows users to specify the resources they want to query; to disallow this (for example, to ensure that people don't use your site's bandwidth to view information about some other site's services), remove the files `$GLOBUS_LOCATION/lib/webmds/conf/openEndedRP` and `$GLOBUS_LOCATION/lib/webmds/conf/openEndedQuery`.

7. Troubleshooting

1. Error handling in WebMDS is currently done by throwing exceptions, which are displayed by Tomcat as stack traces.
2. If you attempt to use WebMDS to collect information from a service that is not running, you will see a stack trace that begins with:

```
org.globus.mds.webmds.xmlSources.resourceProperties.ResourcePropertySourceException: ;
java.net.ConnectException: Connection refused
```

3. When WebMDS sends resource property queries to a secure WSRF service instance (such as an MDS4 Index Server), the WebMDS server must trust the certificate authority that issued the certificate used by the WSRF service instance. If the WebMDS server does not trust the CA used by the remote service, then WebMDS queries will produce a stack trace that includes the following:

```
faultString: org.globus.common.ChainedIOException: Authentication failed
[Caused by: Failure unspecified at GSS-API level [Caused by: Unknown CA]]
```

This can be solved by configuring the Tomcat server that hosts WebMDS to trust the appropriate CA, by either

- placing the CA certificate in `/etc/grid-security/certificates`, or
- placing the CA certificate somewhere else, and setting the Tomcat process's `X509_CERT_DIR` system parameter to the directory in which the CA certificate was installed. One way to do this is to set the `CATALINA_OPTS` environment variable and then restart Tomcat:

```
export CATALINA_OPTS=-DX509_CERT_DIR=/path/to/cert/dir
$CATALINA_HOME/bin/shutdown.sh
$CATALINA_HOME/bin/startup.sh
```

4. If the JVM used by Tomcat is configured to use a blocking random-number source, WebMDS connections to secure Index Servers (or other secure WSRF servers) can hang. This is the default configuration for many installations. One solution is to set the *CATALINA_OPTS* environment variable to ensure that Tomcat's JVM will use a non-blocking random-number source:

```
export CATALINA_OPTS=-Djava.security.egd=/dev/urandom
$CATALINA_HOME/bin/shutdown.sh
$CATALINA_HOME/bin/startup.sh
```



Note

If you encounter this problem with WebMDS, you may also encounter [a similar problem](http://www.globus.org/toolkit/docs/4.0/common/javawscore/admin-index.html#s-javawscore-admin-globusstart)⁵ with the Globus container on the same system.

⁵ <http://www.globus.org/toolkit/docs/4.0/common/javawscore/admin-index.html#s-javawscore-admin-globusstart>

Chapter 7. GT 4.0 WS MDS WebMDS: User's Guide

1. Introduction

WebMDS is a web-based interface for viewing formatted information about Grid resources. In the simplest instance, a web server administrator creates an HTML link that causes the WebMDS server to collect and format information that is presented to the user. Users can also use web forms to specify parameters that control what information is collected and how it's presented.

2. Command-line tools

There is no end-user command-line tool for WebMDS.

2.1. Tool description

The command-line tool `webmds-create-context-file` is used to create Tomcat configuration files needed to deploy WebMDS.

2.2. Command syntax

```
webmds-create-context-file [-f] tomcat_context_file
```

The `tomcat_context_file` argument is the location of the Tomcat configuration file defining the WebMDS context; in a default Tomcat installation, the location of this file will be `$CATALINA_HOME/conf/Catalina/localhost`.

By default, `webmds-create-context-file` will not overwrite an existing context file; the `-f` option is used to force `webmds-create-context-file` to overwrite an existing file.

Note: `webmds-create-context-file` is found in `$GLOBUS_LOCATION/lib/webmds/bin`

2.3. Example

```
$GLOBUS_LOCATION/lib/webmds/bin/webmds-create-context-file -f \  
$CATALINA_HOME/conf/Catalina/localhost
```

2.4. Limitations

Changes to the Tomcat context do not take effect until Tomcat is restarted or reloaded.

3. Graphical user interfaces

3.1. Overview of the purpose and functionality of the GUI

The WebMDS GUI is a web-based interface for browsing formatted XML data, such as the results of resource property queries on a grid service.

3.2. Command and options

WebMDS can be accessed using any web browser. In a default WebMDS installation, the URL `http://host-name:port/webmds` corresponds to the top-level WebMDS web page. This page includes a link to a WebMDS invocation that provides summary information (with links to detailed information) about a locally-running MDS Index server. It also contains a link to a page of sample web forms demonstrating other uses of WebMDS.

3.3. Customizing the web forms used to access WebMDS

The WebMDS servlet is located at `http://your-tomcat-host:your-tomcat-port/webmds/webmds`. It takes the following arguments:

Table 7.1. Form arguments used by WebMDS

<code>info</code>	The name of the XML source that will be used to collect the raw XML data. XML sources are defined by files in <code>\$GLOBUS_LOCATION/lib/webmds/conf</code> . This argument must be specified.
<code>xsl</code>	The name of the XML source that will provide the XSL transform. XML sources are defined by files in <code>\$GLOBUS_LOCATION/lib/webmds/conf</code> . If this argument is not specified, the WebMDS servlet will display raw, untransformed XML.
<code>xml-Source.info_name.param.source_specific_options</code>	Any additional options recognized by the <i>info_name</i> XML source (<i>info_name</i> must be the value of the <i>info</i> argument for this request). Source-specific options are discussed in the next section.
<code>xml-Source.xsl_name.param.source_specific_options</code>	Any additional options recognized by the <i>xsl_name</i> XML source (<i>xsl_name</i> must be the value of the <i>xsl</i> argument for this request). Source-specific options are discussed in the next section.

3.4. Limitations

Error conditions (such as typographical errors in resource property names) are presented as stack traces, rather than user-friendly error messages.

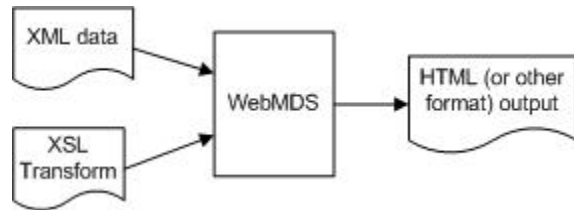
4. Troubleshooting

The commonly-used WebMDS plugins do resource property queries; the Globus Toolkit `wsrf-query` can be used to determine whether the desired information is available directly from the resource.

Chapter 8. GT 4.0 WS MDS WebMDS: Developer's Guide

1. Introduction

WebMDS is a web-based interface for viewing formatted information about Grid resources. Information is collected via a plugin interface and then formatted using an XSLT transform.



2. Before you begin

2.1. Feature summary

Features new in release 4.0:

- Extensible plugin interface to support various mechanisms to gather monitoring information and XSLT transforms.
- Plugins to acquire monitoring information via resource property mechanisms.
- Plugin to acquire XSLT transforms by reading from local files.

Other Supported Features

- WebMDS is a new component, so all its features are "new in release 4.0".

Deprecated Features

- None

2.2. Tested platforms

Tested Platforms for WebMDS:

- The WebMDS server has only been tested with Tomcat version 5.0.28; it has been tested on RedHat Linux (i386) and, to a lesser extent, on Windows XP.
- On the client side, WebMDS should be accessible from any web browser on any platform.

2.2.1. Installing WebMDS on Windows

Although the WebMDS server is not officially supported on non-Unix platforms, and no Windows installer exists for WebMDS, it is possible to run WebMDS on Windows. The following instructions describe how to install WebMDS on a Windows platform.

1. Install [Tomcat](#)¹ and set your CATALINA_HOME environment variable to the directory into which Tomcat was installed.
2. Install the Globus Java WS-Core distribution from the [Globus Toolkit download page](#)². Set your GLOBUS_LOCATION environment variable to the directory into which you installed Globus Java WS-Core
3. Check the ws-mds distribution out of the [Globus CVS repository](#)³, using the globus_4_0_branch tag.
4. Install the servicegroup package:

```
cd c:\wherever\ws-mds\servicegroup\schema
ant deploy
cd ..\source
ant deploy
```

where *wherever* is the directory into which you checked out the ws-mds sources.

5. Install WebMDS:

```
cd c:\wherever\ws-mds\webmds
ant deploy
```

6. Create the webmds context file (this tells Tomcat where to find WebMDS):

```
%GLOBUS_LOCATION%\lib\webmds\bin\webmds-create-context-file %CATALINA_HOME%\conf\Catali
```

7. Restart Tomcat.

WebMDS can then be configured and used as described in the rest of the [WebMDS documentation](#)⁴.

2.3. Backward compatibility summary

Protocol changes since GT version 3.2:

- WebMDS did not exist in GT version 3.2.

API changes since GT version 3.2:

- WebMDS did not exist in GT version 3.2.

Exception changes since GT version 3.2:

- WebMDS did not exist in GT version 3.2.

Schema changes since GT version 3.2:

- WebMDS did not exist in GT version 3.2.

¹ <http://jakarta.apache.org/tomcat/>

² <http://www.globus.org/toolkit/downloads/>

³ <http://www.globus.org/toolkit/docs/development/remote-cvs.html>

⁴ [index.html](#)

2.4. Technology dependencies

WebMDS depends on the following GT components:

- Java WS Core

WebMDS depends on the following 3rd party software:

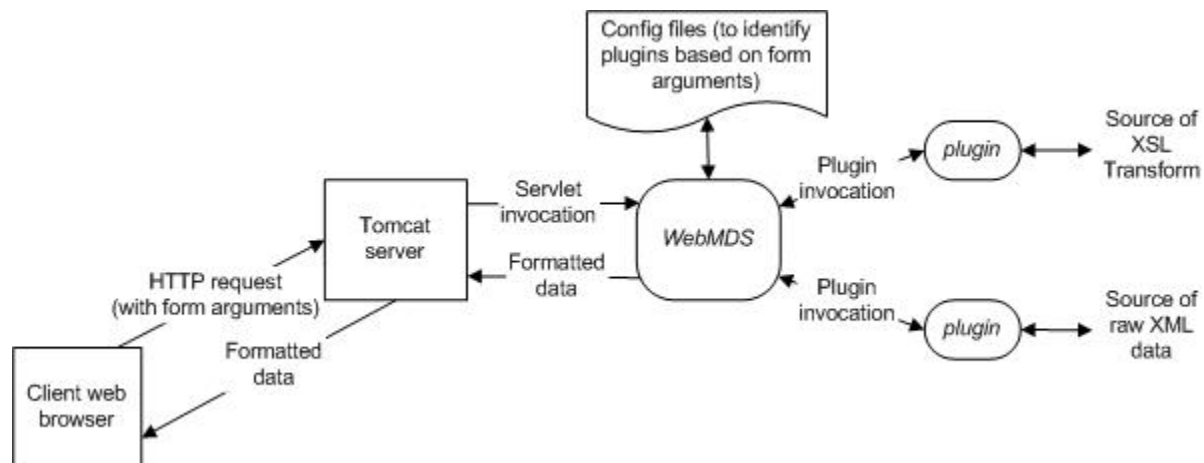
- Tomcat⁵

2.5. Security considerations

By default, the WebMDS plugins distributed as part of the Toolkit do not use authentication credentials -- they retrieve information using anonymous SSL authentication or no authentication at all, and thus retrieve only publicly-available information.

The ResourcePropertyNodeSource and ResourcePropertyQueryNodeSource plugins can be configured either to allow users to specify what resources they want to query or to only allow users to query resources pre-configured by the web administrator. The standard WebMDS deployment allows users to specify the resources they want to query; to disallow this (for example, to ensure that people don't use your site's bandwidth to view information about some other site's services), remove the files \$GLOBUS_LOCATION/lib/webmds/conf/openEndedRP and \$GLOBUS_LOCATION/lib/webmds/conf/openEndedQuery.

3. Architecture and design overview



In a typical WebMDS transaction, a user uses a web browser to send an HTTP request, including some web form arguments, to a web server / servlet container. The web server invokes the WebMDS servlet, which uses the form arguments to determine what plugins to use to retrieve the requested XML data and the XSLT transform to apply to it. The WebMDS servlet passes arguments to the plugins, which then retrieve the appropriate data and XSLT transform. The WebMDS servlet applies the XSLT transformation to the XML data and returns the result to the web server, which sends it back to the client's web browser.

⁵ <http://jakarta.apache.org/tomcat/>

4. Public interface

The semantics and syntax of the APIs and WSDL for the component, along with descriptions of domain-specific structured interface data, can be found in the [Chapter 10, *GT 4.0 Component Guide to Public Interfaces: WS MDS WebMDS*](#).

5. Usage scenarios

There is no "client" programmatic interface to WebMDS; clients communicate using HTTP requests. The web form arguments recognized by WebMDS are documented in [Chapter 7, *GT 4.0 WS MDS WebMDS: User's Guide*](#).

5.1. Creating a new plugin

To create a new plugin to collect raw XML data, write a Java class that implements the `WebmdsXmlSource` or `WebmdsNodeSource` interface. These are documented in [Section 1, "Semantics and syntax of APIs"](#). The `FileXmlSource` and `NodeXmlSource` classes distributed with WebMDS are examples of classes that implement `WebmdsXmlSource`; the `ResourcePropertyNodeSource` and `ResourcePropertyQueryNodeSource` classes distributed with WebMDS are examples of classes that implement the `WebmdsNodeSource` interface.

5.2. Changing format of output

To change the appearance of the output of WebMDS, create a new XSLT transform; see the [W3C XSLT Documentation](#)⁶ for more information.

6. Troubleshooting

Log information from WebMDS and any WebMDS plugins will be logged by the servlet container into which WebMDS has been deployed. In a vanilla Tomcat 5.0.28 distribution, this information will show up in the file `$CATALINA_HOME/logs/catalina.out`

7. Related Documentation

None available at this time.

⁶ <http://www.w3.org/TR/xslt>

Chapter 9. GT 4.0 Component Fact Sheet: WS MDS WebMDS

1. Brief component overview

WebMDS enables end users to view monitoring information via a standard web browser interface, without installing any additional software on their PC. WebMDS is implemented as a servlet that uses a plugin interface to gather monitoring information (or any other information in XML format) and XSLT transforms, and present the data to the user in a readable form. Web site administrators can customize their own WebMDS deployments by using HTML form options, configuring different plugins to collect data and XSLT transforms, and creating their own plugins and XSLT transforms.

2. Summary of features

Features new in release 4.0:

- Extensible plugin interface to support various mechanisms to gather monitoring information and XSLT transforms.
- Plugins to acquire monitoring information via resource property mechanisms.
- Plugin to acquire XSLT transforms by reading from local files.

Other Supported Features

- WebMDS is a new component, so all its features are "new in release 4.0".

Deprecated Features

- None

3. Usability summary

This section does not apply for WebMDS as it is a new component with this release.

4. Backward compatibility summary

Protocol changes since GT version 3.2:

- WebMDS did not exist in GT version 3.2.

API changes since GT version 3.2:

- WebMDS did not exist in GT version 3.2.

Exception changes since GT version 3.2:

- WebMDS did not exist in GT version 3.2.

Schema changes since GT version 3.2:

- WebMDS did not exist in GT version 3.2.

5. Technology dependencies

WebMDS depends on the following GT components:

- Java WS Core

WebMDS depends on the following 3rd party software:

- [Tomcat](#)¹

6. Tested platforms

Tested Platforms for WebMDS:

- The WebMDS server has only been tested with Tomcat version 5.0.28; it has been tested on RedHat Linux (i386) and, to a lesser extent, on Windows XP.
- On the client side, WebMDS should be accessible from any web browser on any platform.

6.1. Installing WebMDS on Windows

Although the WebMDS server is not officially supported on non-Unix platforms, and no Windows installer exists for WebMDS, it is possible to run WebMDS on Windows. The following instructions describe how to install WebMDS on a Windows platform.

1. Install [Tomcat](#)² and set your CATALINA_HOME environment variable to the directory into which Tomcat was installed.
2. Install the Globus Java WS-Core distribution from the [Globus Toolkit download page](#)³. Set your GLOBUS_LOCATION environment variable to the directory into which you installed Globus Java WS-Core
3. Check the ws-mds distribution out of the [Globus CVS repository](#)⁴, using the globus_4_0_branch tag.
4. Install the servicegroup package:

```
cd c:\wherever\ws-mds\servicegroup\schema
ant deploy
cd ..\source
ant deploy
```

where *wherever* is the directory into which you checked out the ws-mds sources.

5. Install WebMDS:

```
cd c:\wherever\ws-mds\webmds
ant deploy
```

¹ <http://jakarta.apache.org/tomcat/>

² <http://jakarta.apache.org/tomcat/>

³ <http://www.globus.org/toolkit/downloads/>

⁴ <http://www.globus.org/toolkit/docs/development/remote-cvs.html>

6. Create the webmds context file (this tells Tomcat where to find WebMDS):

```
%GLOBUS_LOCATION%\lib\webmds\bin\webmds-create-context-file %CATALINA_HOME%\conf\Catali
```

7. Restart Tomcat.

WebMDS can then be configured and used as described in the rest of the [WebMDS documentation](#)⁵.

7. Associated standards

Associated standards for WS MDS WebMDS:

- HyperText Transfer Protocol (HTTP)
- HyperText Markup Language (HTML)
- XSL Transformations (XSLT)
- WebMDS is implemented as a Java Servlet

8. For More Information

Click [here](#)⁶ for more information about this component.

⁵ [index.html](#)

⁶ [index.html](#)

Chapter 10. GT 4.0 Component Guide to Public Interfaces: WS MDS WebMDS

1. Semantics and syntax of APIs

1.1. Programming Model Overview

There is no "client" API for accessing WebMDS; WebMDS is a servlet that is accessed via web forms.

WebMDS uses a *WebMDS plugin* (a Java class that implements the `WebmdsXmlSource` interface) to acquire XML documents (which can be used either as raw information sources or as XSL transformations). WebMDS comes with two WebMDS plugins: `FileXmlSource`, which reads XML from a file (and is primarily used to acquire XSL transformations), and `NodeXmlSource`. `NodeXmlSource` in turn calls a *node source plugin* (a Java class that implements the `WebmdsNodeSource` interface) to acquire an XML DOM document. acquires XML information using a *WebMDS XML source*, a Java class that implements the `WebmdsXmlSource` interface. To summarize:

- WebMDS is a servlet that uses plugins to acquire XML documents containing raw data and XSL transformations, and then applies the acquired XSL transformation on the acquired data.
 - The plugins used by WebMDS implement the `org.globus.mds.webmds.WebmdsXmlSource` interface.
 - WebMDS plugins include:
 - `org.globus.mds.webmds.xmlSources.file.FileXmlSource`, which reads XML from a file, and
 - `org.globus.mds.webmds.xmlSources.xmlDomNode.NodeXmlSource`, which uses its own plugin interface to acquire XML DOM documents.
 - The plugins used by `NodeXmlSource` implement the `org.globus.mds.webmds.xmlSources.xmlDomNode.WebmdsNodeSource` interface
 - Node source plugins include `org.globus.mds.webmds.xmlSources.resourceProperties.ResourcePropertyNodeSource` and `org.globus.mds.webmds.xmlSources.resourceProperties.ResourcePropertyQueryNodeSource`, which acquire resource property information.
 - The raw XML data acquired by WebMDS is processed by XSL transformations; see the [W3C XSLT Documentation](http://www.w3.org/TR/xslt)¹ for more information on creating XSL transforms.

1.2. Component API

- [Core WebMDS documentation](#)² (includes the WebMDS servlet and the `WebmdsNodeSource` interface)
- [FileXMLSource documentation](#)³

¹ <http://www.w3.org/TR/xslt>

² http://www.globus.org/api/javadoc-4.0.0/globus_wsrf_mds_webmds/

³ http://www.globus.org/api/javadoc-4.0.0/globus_wsrf_mds_webmds_file_source/

- [NodeXmlSource documentation](#)⁴ (including the `WebmdsNodeSource` interface)
- [Resource property node source plugins](#)⁵.

2. Semantics and syntax of the WSDL

2.1. Protocol overview

WebMDS is not a Web service and does not have any associated WSDL. For information on the web form-based protocol used by WebMDS, see the [documentation](#)⁶ on the WebMDS Graphical User Interface.

3. Command-line tools

There is no end-user command-line tool for WebMDS.

3.1. Tool description

The command-line tool `webmds-create-context-file` is used to create Tomcat configuration files needed to deploy WebMDS.

3.2. Command syntax

```
webmds-create-context-file [-f] tomcat_context_file
```

The `tomcat_context_file` argument is the location of the Tomcat configuration file defining the WebMDS context; in a default Tomcat installation, the location of this file will be `$CATALINA_HOME/conf/Catalina/localhost`.

By default, `webmds-create-context-file` will not overwrite an existing context file; the `-f` option is used to force `webmds-create-context-file` to overwrite an existing file.

Note: `webmds-create-context-file` is found in `$GLOBUS_LOCATION/lib/webmds/bin`

3.3. Example

```
$GLOBUS_LOCATION/lib/webmds/bin/webmds-create-context-file -f \  
$CATALINA_HOME/conf/Catalina/localhost
```

3.4. Limitations

Changes to the Tomcat context do not take effect until Tomcat is restarted or reloaded.

⁴ http://www.globus.org/api/javadoc-4.0.0/globus_wsrf_mds_webmds_xml_dom_source/

⁵ http://www.globus.org/api/javadoc-4.0.0/globus_wsrf_mds_webmds_resource_property_source/

⁶ <http://www.globus.org/toolkit/docs/4.0/info/webmds/>

4. Overview of Graphical User Interface

4.1. Overview of the purpose and functionality of the GUI

The WebMDS GUI is a web-based interface for browsing formatted XML data, such as the results of resource property queries on a grid service.

4.2. Command and options

WebMDS can be accessed using any web browser. In a default WebMDS installation, the URL `http://host-name:port/webmds` corresponds to the top-level WebMDS web page. This page includes a link to a WebMDS invocation that provides summary information (with links to detailed information) about a locally-running MDS Index server. It also contains a link to a page of sample web forms demonstrating other uses of WebMDS.

4.3. Customizing the web forms used to access WebMDS

The WebMDS servlet is located at `http://your-tomcat-host:your-tomcat-port/webmds/webmds`. It takes the following arguments:

Table 10.1. Form arguments used by WebMDS

<code>info</code>	The name of the XML source that will be used to collect the raw XML data. XML sources are defined by files in <code>\$GLOBUS_LOCATION/lib/webmds/conf</code> . This argument must be specified.
<code>xsl</code>	The name of the XML source that will provide the XSL transform. XML sources are defined by files in <code>\$GLOBUS_LOCATION/lib/webmds/conf</code> . If this argument is not specified, the WebMDS servlet will display raw, untransformed XML.
<code>xml-Source.info_name.param.source_specific_options</code>	Any additional options recognized by the <i>info_name</i> XML source (<i>info_name</i> must be the value of the <i>info</i> argument for this request). Source-specific options are discussed in the next section.
<code>xml-Source.xsl_name.param.source_specific_options</code>	Any additional options recognized by the <i>xsl_name</i> XML source (<i>xsl_name</i> must be the value of the <i>xsl</i> argument for this request). Source-specific options are discussed in the next section.

4.4. Limitations

Error conditions (such as typographical errors in resource property names) are presented as stack traces, rather than user-friendly error messages.

5. Semantics and syntax of domain-specific interface

5.1. Interface introduction

WebMDS uses a web form interface to specify parameters such as where to find raw data and what XSLT transformations to apply to that data.

5.2. Syntax of the interface

The web form interface is described in [Section 3, “Graphical user interfaces”](#)

6. Configuration interface

6.1. Configuration overview

WebMDS can be configured to get information from any of various sources and to filter it through any XSL transform. WebMDS uses configuration files to specify the location of (and to name) sources of information and xsl and web form arguments to select among these configured information sources and xsl transforms.

By default, WebMDS comes configured to report information about an index server using transaction-level security on the default port (8443) on the local system. If you are running the Globus Toolkit in this default configuration, then you can use WebMDS to query your local *Index Service* without any configuration changes.

If you wish to monitor a different index server, you will need to edit the file `$GLOBUS_LOCATION/lib/webm-ds/conf/indexinfo` to change the URL in the line:

```
<value>https://127.0.0.1:8443/wsrf/services/DefaultIndexService</value>
```

to match the URL of your default index service. Changes to WebMDS configuration files take effect the next time that Tomcat is restarted.

For other configuration changes (e.g., monitoring different kinds of services), see the detailed configuration information below.

6.2. Syntax of the interface

Each configuration file in `$GLOBUS_LOCATION/lib/webm-ds/conf` defines a source of XML, which can be used in an HTML form to specify sources of information and XSL transforms. The distribution contains some standard configuration files in this directory, including:

Table 10.2. Pre-configured information sources

indexinfo	all resource properties from an index server running with transaction-level security on port 8443 on the local host
indexinfo_nosec	all resource properties from an index server running with no security on port 8080 on the local host
openEndedQuery	all resource properties from a user-specified grid service
openEndedRP	a user-specified resource property from a user-specified grid service
servicegroupxsl	an xsl transform that presents summary information about a service group
sgedetail	an xsl transform that presents detailed information about a service group entry

Each configuration file defines a `WebmdsConfig` object. A `WebmdsConfig` object consists of:

- A `description`: a textual description of the XML source being defined.
- A `className`: the name of the Java class that will be used to acquire the XML data.
- Zero or more `parameter` objects, each of which consists of the name of some parameter recognized by the Java class specified by `className`, and the string `value` of that parameter.

For example, this is `$GLOBUS_LOCATION/lib/webmds/conf/servicegroupxsl`, which defines the `servicegroupxsl` XML source:

```
<WebmdsConfig>
  <description>
    XSL file to show service group summary information
  </description>
  <className>org.globus.mds.webmds.xmlSources.file.FileXmlSource</className>
  <parameter>
    <name>file</name>
    <value>xslfiles/servicegrouptable.xsl</value>
  </parameter>
</WebmdsConfig>
```

This file tells WebMDS to use the `org.globus.mds.webmds.xmlSources.file.FileXmlSource` Java class (a class which reads XML from a local file) to collect XML data and to pass a `file` parameter (which that Java class interprets as the name of the file to open, relative to the WebMDS base directory).

Tomcat must be restarted (or one of the more advanced Tomcat administrative mechanisms must be used) for changes to these configuration files to take effect.

6.3. XML Sources included with WebMDS

6.3.1. FileXMLSource

The class `org.globus.mds.webmds.xmlSources.file.FileXmlSource` reads XML from a file, and recognizes a single parameter:

Table 10.3. Configuration parameters used with FileXMLSource

<code>file</code>	The name of the file to read. Relative path names are interpreted relative to the WebMDS base directory (<code>\$GLOBUS_LOCATION/lib/webmds</code>).
-------------------	--

6.3.2. NodeXMLSource

This XML source class uses a `WebmdsNodeSource` object to fetch an XML document and return it in a form that is usable by WebMDS. It recognizes the following options:

Table 10.4. Configuration parameters used with NodeXMLSource

<code>class</code>	The name of a class that implements the <code>WebmdsNodeSource</code> interface. An instance of this class will be used to get an XML document.
<code>parameters</code>	Additional parameters are passed to an instance of the class specified by the <code>class</code> argument.

6.3.3. Classes That Implement WebmdsNodeSource

The following classes implement the `NodeXMLSource` interfaces and can be used in conjunction with `NodeXMLSource`

6.3.4. ResourcePropertyQueryNodeSource

This class performs a resource property query to get all the resource properties for some web service. It recognizes the following configuration parameters:

Table 10.5. Configuration parameters used with ResourcePropertyQueryNodeSource

<code>endpoint</code>	The endpoint name to be used in a resource property query.
<code>endpointKeyName</code> and <code>endpointKeyValue</code>	An optional key/value pair to use as reference properties for the endpoint specified with the <code>endpoint</code> parameter.
<code>allowUserEndpoints</code>	If true, values for <code>xmlSource.sourceName.param.endpoint</code> , <code>xmlSource.sourceName.param.endpointKeyName</code> , and <code>xmlSource.sourceName.param.endpointKeyValue</code> specified in the request will override the configured endpoint value.
<code>endpointFile</code>	The name of a file from which the endpoint information (in XML) will be read. This configuration parameter can never be overridden by request arguments.

6.3.5. ResourcePropertyNodeSource

This class queries a web service for a single resource property. It recognizes the following parameters:

Table 10.6. Configuration parameters used with ResourcePropertyNodeSource

endpoint	The endpoint name to be used in a resource property query.
endpointKeyName and endpointKeyValue	An optional key/value pair to use as reference properties for the endpoint specified with the endpoint parameter.
allowUserEndpoints	If true, values for <code>xmlSource.sourceName.param.endpoint</code> , <code>xmlSource.sourceName.param.endpointKeyName</code> , and <code>xmlSource.sourceName.param.endpointKeyValue</code> specified in the request will override the configured endpoint value.
endpointFile	The name of a file from which the endpoint information (in XML) will be read. This configuration parameter can never be overridden by request arguments.
rpNamespace	The namespace part of the QName of the resource property to be queried for.
rpName	The local name part of the QName of the resource property to be queried for.
allowUserResourceProperties	If true, values of <code>xmlSource.sourceName.param.rpNamespace</code> and <code>xmlSource.sourceName.param.rpNames</code> specified in the request will override the configured resource property namespace and name.

7. Environment variable interface

WebMDS does not require that any environment variables be set by the client or by the Tomcat server.

Chapter 11. GT 4.0 WS MDS WebMDS: Quality Profile

1. Test coverage reports

- None available at this time.

2. Code analysis reports

- None available at this time.

3. Outstanding bugs

- [3040: Webmds can break if started from the wrong directory](#)¹
- [3051: Handle huge indexes](#)²
- [3160: Format summary line for RLS](#)³
- [All open bug reports and enhancement requests for WebMDS](#)⁴

4. Bug Fixes

- [2347: Information missing from service group entry detail page in some cases](#)⁵
- [2275: Misleading summary information in servicegrouptable.xml](#)⁶
- [2257: Add RFT support into servicegroup table XSL](#)⁷
- [2143: stylesheet that displays simple table](#)⁸
- [2806: Fix default values in "open-ended resource property query" sample form](#)⁹
- [2769: webmds has missing files in filelists](#)¹⁰

¹ http://bugzilla.globus.org/globus/show_bug.cgi?id=3040

² http://bugzilla.globus.org/globus/show_bug.cgi?id=3051

³ http://bugzilla.globus.org/globus/show_bug.cgi?id=3160

⁴ http://bugzilla.globus.org/globus/buglist.cgi?short_desc_type=allwordssubstr&short_desc=&product=MDS&component=wsrf_webm-ds&long_desc_type=allwordssubstr&long_desc=&bug_file_loc_type=allwordssubstr&bug_file_loc=&bug_status=NEW&bug_status=AS-SIGNED&bug_status=REOPENED&emailtype1=substring&email1=&emailtype2=substring&email2=&bugidtype=include&bug_id=&votes=&chan-geidin=&chfieldfrom=&chfieldto=Now&chfieldvalue=&cmdtype=doit&newqueryname=&order=Reuse+same+sort+as+last+time&field0-0-0=noop&type0-0-0=noop&value0-0-0=

⁵ http://bugzilla.globus.org/globus/show_bug.cgi?id=2347

⁶ http://bugzilla.globus.org/globus/show_bug.cgi?id=2275

⁷ http://bugzilla.globus.org/globus/show_bug.cgi?id=2257

⁸ http://bugzilla.globus.org/globus/show_bug.cgi?id=2143

⁹ http://bugzilla.globus.org/globus/show_bug.cgi?id=2806

¹⁰ http://bugzilla.globus.org/globus/show_bug.cgi?id=2769

- All fixed bugs and enhancement requests for WebMDS¹¹

5. Performance reports

- None available at this time.

¹¹ http://bugzilla.globus.org/globus/buglist.cgi?short_desc_type=allwordssubstr*&short_desc=&product=MDS&component=wsrf_web-mds&long_desc_type=allwordssubstr*&long_desc=&bug_file_loc_type=allwordssubstr*&bug_file_loc=&bug_status=RESOLVED&bug_status=VERIFIED&bug_status=CLOSED&emailtype1=substring*&email1=&emailtype2=substring*&email2=&bugidtype=include*&bug_id=&votes=&changedin=&chfieldfrom=&chfieldto=Now*&chfieldvalue=&cmdtype=doit*&newqueryname=&order=Reuse+same+sort+as+last+time*&field0-0-0=noop*&type0-0-0=noop*&value0-0-0=

GT 4.0 WS MDS Glossary

A

Aggregator Framework	A software framework used to build services that collect and aggregate data. MDS4 Services (such as the Index and Trigger services) are built on the Aggregator Framework, and are sometimes called Aggregator Services.
aggregator services	Services that are built on the Aggregator Framework, such as the MDS4 Index Service and Trigger Service. See Also Aggregator Framework , Index Service , Trigger Service .
aggregator source	A Java class that implements an interface (defined as part of the Aggregator Framework) to collect XML-formatted data. MDS4 contains three aggregator sources: the query aggregator source, the subscription aggregator source, and the execution aggregator source. See Also query aggregator source , subscription aggregator source , execution aggregator source .

E

execution aggregator source	An Aggregator Source (included in MDS4) that executes an administrator-supplied program to collect information and make it available to an Aggregator Service such as the Index Service. See Also aggregator source .
-----------------------------	--

G

Ganglia	A cluster monitoring tool. See http://ganglia.sourceforge.net .
---------	--

H

Hawkeye	A monitoring service for Condor Pools. See http://www.cs.wisc.edu/condor/hawkeye/ .
---------	--

I

Index Service	An aggregator service that serves as a registry similar to UDDI, but much more flexible. Indexes collect information and publish that information as WSRF resource properties. See Also aggregator services .
information provider	A "helper" software component that collects or formats resource information, for use by an aggregator source or by a WSRF service when creating resource properties.

Q

query aggregator source	An aggregator source (included in MDS4) that polls a WSRF service for resource property information. See Also aggregator source .
-------------------------	--

S

subscription aggregator source	An aggregator source (included in MDS4) that collects data from a WSRF service via WSRF subscription/notification. See Also aggregator source .
--------------------------------	--

T

Trigger Service	An aggregator service that collects information and compares that data against a set of conditions defined in a configuration file. When a condition is met, or triggered, an action takes place, such as emailing a system administrator when the disk space on a server reaches a threshold. See Also aggregator services .
-----------------	--

W

WebMDS	A web-based interface to WS-RF resource property information that can be used as a user-friendly front-end to the Index Service or other WS-RF services.
--------	--