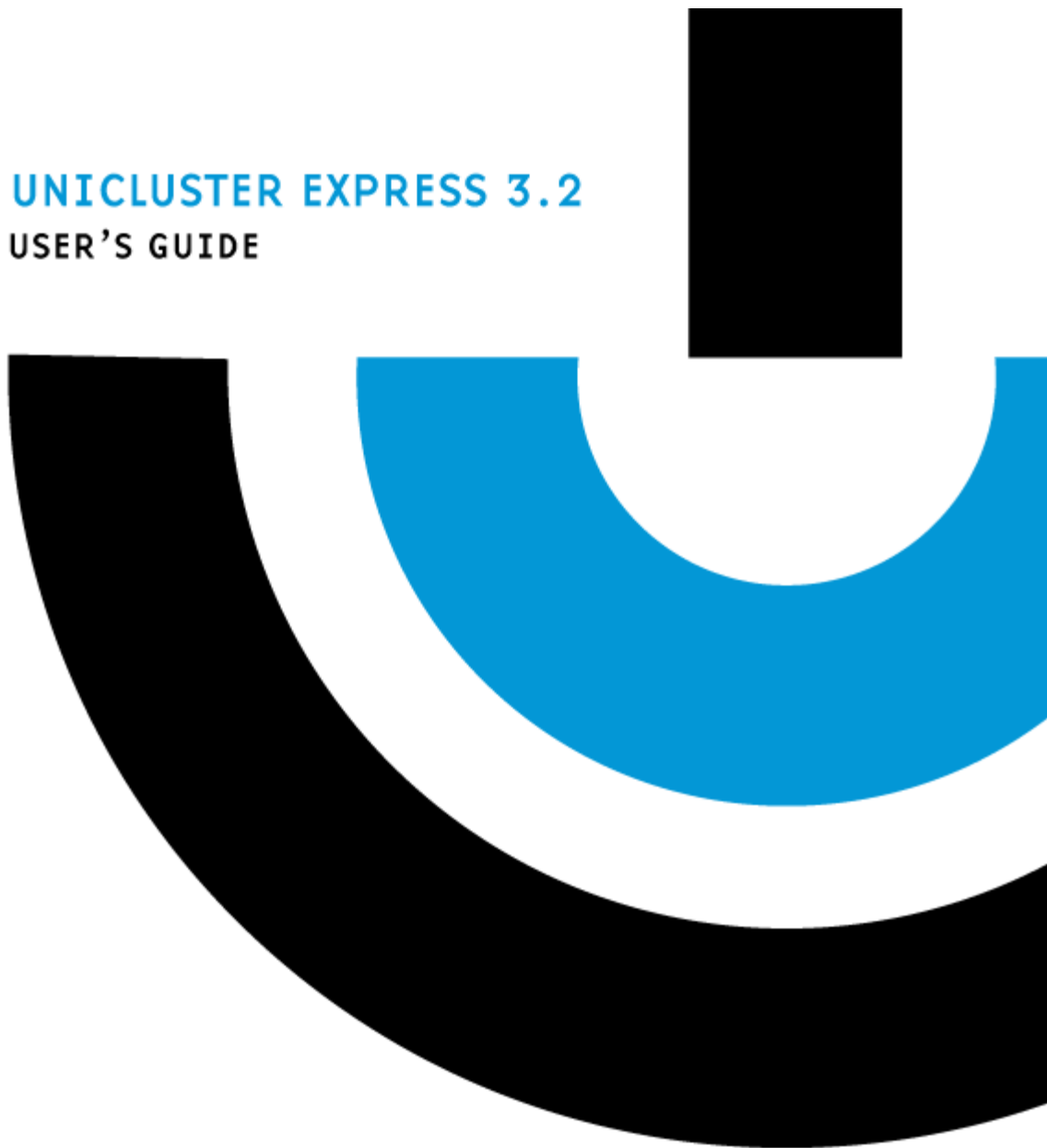


UNICLUSTER EXPRESS 3.2

USER'S GUIDE



March 2008

Univa UD Inc.
<http://www.univaud.com>

Copyright, Trademark, and Version Notices

Copyright Notice

Copyright © 2000-2008 Univa UD, Inc. All rights reserved.

Information relating to the architecture and software described in this guide is proprietary to Univa UD and, as applicable, its licensors. Rights to use the software and the accompanying documentation are governed by a separate license agreement from Univa UD presented prior to software installation.

Information in this document is subject to change without notice. No part of this guide may be reproduced or published in any form or by any means, electronic or mechanical, including photocopying, for any purpose other than the purchaser's personal use.

Trademark Notice

Univa UD and UniCluster Express are trademarks or service marks of Univa UD, Inc. Other company and product names may be the trademarks or service marks of third parties. Third party software is installed as distinct, unmodified software programs during the UniCluster Express platform installation process.

Licensing

UniCluster Express includes software components produced by Univa UD and a number of other third parties. The list of components and their associated licenses is provided with the software.

The source code of each UniCluster Express component is copyright of the original publishers, and use of the source code of each component is governed by the terms and conditions of the license provided with that component by the original publishers. Source code for most UniCluster Express open source components is available at the Grid.org website. However, some third party components which UniCluster Express uses unmodified are available via grid.org only as object code (e.g. jar files), in which case source can be found at that component's open source project site.

The UniCluster Express object code is governed by the same terms and conditions as the source code from which that object code is derived. The exception to this is the Grid Engine component, for which UniCluster Express uses the Sun Grid Engine object code provided by Sun Microsystems, which is built from the Grid Engine open source but is distributed with a different object code license.

Contents

Preface	v
Chapter 1: Logging on to the Cluster	1
To log in to the cluster	1
Using unicluster-grid-logon	2
Chapter 2: Using Grid Engine	5
Submitting and Managing Jobs	5
Using qsub	5
Using qstat	6
Using qdel	6
Finding Queues and Hosts	7
Using qconf	7
Chapter 3: Using Ganglia	9
Using Ganglia Command Line Tools	9
Using gmetric	9
Using "gstat"	10
Chapter 4: Using the Monitor Console	13
Starting the Monitor Console	13
Using a Cluster Workstation	13
Using a Non-Cluster Machine	15
Running the Monitor Console on a Non-Cluster Machine	15
Changing Monitor Console Preferences	17
Changing Displayed Time Ranges	20
Filtering Information	29
Saving and Organizing Bookmarks	30
Quitting the Monitor Console	38
Chapter 5: Using GridFTP	39
Using globus-url-copy	39

Chapter 6: Using WS-GRAM	43
Using globusrun-ws	43
Chapter 7: Using GSI-OpenSSH	49
Using GSI-OpenSSH Command Line Tools	49
Using gsissh	49
Using gsiscp	50
Using gsisftp	51
Interactive Commands used with gsisftp	51
Chapter 8: Using Reliable File Transfer	53
Using rft	54
Using rft-delete	56
Chapter 9: Troubleshooting	59
WSRF Tools	61
Using "wsrf-destroy"	61
Using "wsrf-set-termination-time"	62
Using "wsrf-query"	64
Using "wsrf-get-property"	65
Using "wsrf-get-properties"	66
Using "wsrf-insert-property"	68
Using "wsrf-update-property"	69
Using "wsrf-delete-property"	71
WS-Notification Tools	72
Using "wsn-get-current-message"	72
Using "wsn-pause-subscription"	74
Using "wsn-resume-subscription"	75
Using "wsn-subscribe"	76
Glossary	79

Preface

About This UniCluster Express

UniCluster Express 3.2 is a free cluster management software product that integrates best-of-breed technologies to provide everything needed to run cluster applications, including job scheduling, monitoring, security, and remote access. Key UniCluster Express benefits include:

- A dramatic reduction in software costs by using open source solutions, with increased resource utilization, user productivity, and processing efficiency of jobs.
- A dramatic reduction in ongoing management and support costs by using of best-of-breed management, monitoring and scheduling software, coupled with an ongoing, comprehensive, integrated support offering.
- Extending real-time cluster availability to remote users securely and reliably, with the added benefits of optimized availability for all users, automated resource allocation, and extensive security protection.
- Fewer error-prone manual installation and configuration processes because ad-hoc integration and maintenance are eliminated by using a single, best-of-breed solution.
- Reduced system downtime because management and monitoring tools, coupled with extensive testing and enhanced integration, offer significant time savings for installation, configuration, troubleshooting, integration, and maintenance tasks.
- A seamless migration path to UniCluster Express Professional and Grid MP, so your cluster can quickly and efficiently scale as your business needs expand.

UniCluster Express is the result of an extensive evaluation of all leading open source cluster technology components, integrating only the components with the greatest reliability, highest long term vitality, and best laid product paths into a single bundled offering.

Part of the extensive evaluation process included developing relationships with the software components' architects and developers. This ensures the comprehensive support options for UniCluster Express are comprehensive and complete.

About this Guide

This guide gives the prerequisites for installation and guides you through the installation process. This book contains the following chapters:

-
- ‘[Chapter 1,Logging on to the Cluster](#),’ on page 1
 - ‘[Chapter 2,Using Grid Engine](#),’ on page 5
 - ‘[Chapter 3,Using Ganglia](#),’ on page 9
 - ‘[Chapter 4,Using the Monitor Console](#),’ on page 13
 - ‘[Chapter 5,Using GridFTP](#),’ on page 39
 - ‘[Chapter 6,Using WS-GRAM](#),’ on page 43
 - ‘[Chapter 8,Using Reliable File Transfer](#),’ on page 53
 - ‘[Chapter 9,Troubleshooting](#),’ on page 59
 - ‘[Chapter ,WSRF and WNS Command Line Tools](#),’ on page 61
 - ‘[Glossary](#),’ on page 79, presents Univa UD-, industry-, or field-specific terms.

Audience

This document is intended for UniCluster Express end users. It is not intended as a tutorial on clusters or grid computing. For more information, please see the Univa UD website, <http://www.univaud.com>.

Related documentation

Please refer to the following resources for additional information:

- UniCluster Express User’s Guide
- UniCluster Express Administrator’s Guide
- The Univa UD website, <http://www.univaud.com>, which has information about Univa UD, product support options, and whitepapers on grid and cluster technology.
- Grid.org, <http://www.grid.org>, which has user forums, downloads of the UniCluster Express 3.2 installer and documentation, and a wiki for additional configuration and use information.

Using this Guide

Typographical Conventions

The following table shows the typographical conventions used in this book.

Table 1. Typographical conventions

Convention	Description
Courier	Code examples, file and pathnames, screen output.
Bold Courier	Functions, commands, text that must be typed literally.
<i>Bold Italic Courier</i>	Function and command variables; these are replaced with appropriate installation-specific values, such as a username.
Bold Times New Roman	GUI menu commands, menu options, buttons.
Bold Times New Roman	GUI screen names, panes, pop-up menu names.

Viewing this Guide

This document is best viewed with Adobe® Acrobat® Reader® version 5.0 or later. Adobe Reader is available as a free download at <http://www.adobe.com/products/acrobat/readstep2.html>

This document contains a number of features that enable you to navigate quickly and easily through the guide:

- Bookmarks on the left side of the screen allow you to jump to a section easily. To use bookmarks, click the bookmark for the section. To return, right-click and select **Go To Previous View** from the pop-up menu.
- Links to figures and sections can be seen throughout the document. These links are formatted in blue, underlined text for easy identification and allow you to jump to a specific figure, table, or section referenced in the text. As with bookmarks, to return to the previous view, right-click and select **Go To Previous View** from the pop-up menu.

Software maintenance and upgrades

Univa UD is committed to improving continuously its products and services. When new maintenance or product releases are available, Univa UD Professional Services will contact you to help plan your upgrade.

Upgrades involve:

- assessing the application impact
- preparing required database migration scripts
- testing upgrade procedures prior to implementation

Some licensing issues may also apply. For more information about upgrades, contact Univa UD Professional Services.

Contacting Univa UD

For technical support, or to inquire about the information contained in this document, contact Univa UD in one of the following ways:

- For technical support or training, send e-mail to support@univaud.com.
- For information on support options, visit the website at <http://www.univaud.com/> for more information.
- Support can also be reached at telephone +1.800.370.5320, between 9:00 a.m. and 5:00 p.m. (CST), Monday through Friday.

About Grid.org

UniCluster Express also has on line community support through Grid.org, <http://www.grid.org/>.

Grid.org is a community for open source cluster and grid users, administrators and developers. Joining Grid.org allows you to interact with other cluster and grid users, participate in forums and blogs, access whitepapers and case studies, explore upcoming events, and download open source software for integrated cluster management.

Grid.org is intended to provide a single location where open-source cluster and grid information can be aggregated and where community members can exchange information, experiences, and ideas related to the complete open source cluster software stack.

On Grid.org, you'll find a variety of communication methods, including wikis, mailing lists, forums, and blogs. There is also public access to a Bugzilla tracking system.

To access various Grid.org features, use the following links:

- Grid.org home page, <http://www.grid.org/>

-
- Product and Documentation download page, <http://www.grid.org/download>
 - Issue Tracking, <https://bugs.grid.org/>
 - Forums, <http://www.grid.org/forum/>
 - Wiki, http://www.grid.org/wiki/index.php?title=Main_Page
 - Mailing list, <http://www.grid.org/mailling-list>
 - Registration, <http://www.grid.org/forum/register.php>

Chapter 1: Logging on to the Cluster

Before using any web service, Grid Engine or Ganglia, you must log in to the cluster and obtain a proxy for your work session. Once you are logged in, you may use GridFTP to move files, submit jobs through WS-GRAM or Grid Engine, or monitor jobs using Ganglia.

- | | |
|-------------|--|
| NOTE | <ul style="list-style-type: none">• You should log in to the cluster with your regular UNIX username and password issued to you by your system administrator; security is based on your UNIX identity• Throughout this chapter, <i>install-dir</i> refers to the UniCluster Express installation location. The environmental variable, <code>\$GLOBUS_LOCATION</code>, points to the UniCluster Express installation directory and is used by Globus Toolkit components. Where required, you may set this environmental variable, or enter the complete path to the UniCluster Express installation location. |
|-------------|--|

You can log in to the cluster from any workstation or logon node.

To log in to the cluster

1. First configure the shell environment by sourcing the appropriate environment setup script:

- `bash` or `sh`> users execute:

```
[user@host ~]$ . <install-dir>/unicluster-user-env.sh
```

- `csh` or `tcsh`< users execute:

```
[user@host ~]$ source <install-dir>/unicluster-user-env.csh
```

where `<install-dir>` is the path to the UniCluster Express installation directory.

NOTE	Does not support the use of <code>zsh</code> shell.
-------------	---

-
2. Log in to the cluster using **unicluster-grid-logon**. This command retrieves your user credential and feeds it to a properly created and protected proxy certificate file which is required to use the cluster tools.

Using unicluster-grid-logon

Syntax for **unicluster-grid-logon** is:

```
unicluster-grid-logon -option ...
```

For example:

```
[user@host ~]$ source &lt;install-dir>/unicluster-user-env.sh
```

```
[user@host ~]$ unicluster-grid-logon
```

The accepted options for unicluster-grid-logon are in the table below.

Table 1. Options for unicluster-grid-logon

<i>Option</i>	Description
<i>-help</i>	Displays tool help.
<i>-verbose</i>	Specifies the verbose output mode.
<i>-dryrun</i>	Shows the user the arguments and parameters that would be executed in an actual unicluster-grid-logon.
<i>-replace</i>	Forces the tool to replace proxy certificate files if files Already exist for the user in the specified location. Used in conjunction with <i>-split</i> option.
<i>-f</i>	Specifies the path to a valid proxy certificate file.
<i>-split</i>	Splits the key and certificate chain into two separate files; must be used in conjunction with <i>-cert</i> and <i>-key</i> options.
<i>-cert</i> <i>/path/to/certfile</i>	Specifies the cert file; valid only when used in conjunction with <i>-split</i> option.
<i>-key</i> <i>/path/to/keyfile</i>	Specifies the key file; valid only when used in conjunction with <i>-split</i> option.
<i>-t hours</i>	Specifies the time, in hours, for a valid log in; maximum time is 168 hours.

If the **-split** option is used with the command **unicluster-grid-logon**, then the command splits the key and certificate chain into two separate files. For example:

```
[user@host ~]$ unicluster-grid-logon -split -cert  
~/usercert.pem -key userkey.pem -t 168
```

```
[user@host ~]$ ll | grep pem
-rw-r--r--  1 user user  989 Oct 26 11:31 usercert.pem
-rw-----  1 user user  887 Oct 26 11:31 userkey.pem
```

Having the ability to generate user certificate and key files is useful when using the Monitor Console on a node that is outside of the cluster. For details, see Chapter 4, “[Using the Monitor Console](#),” on page 13.

By default, user certificate and key files generated by using **-split** option are valid for 12 hours; to generate files that are valid for a longer period of time, use the **-t hours** option.

Chapter 2: Using Grid Engine

This chapter gives information on using Grid Engine and is designed to help first-time Grid Engine users prepare and submit jobs. For complete Grid Engine documentation, please refer to the Grid Engine website, <http://gridengine.sunsource.net/>.

Grid Engine is a job scheduler that features multiple advanced scheduling algorithms that allow powerful, policy-based resource allocation for cluster jobs. Users may run Grid Engine jobs from any submit host; a list of submit hosts is available by running **qconf -ss**; details on using **qconf** are in the section [Finding Queues and Hosts](#).

Throughout this chapter, *install-dir* refers to the UniCluster Express installation location. The environmental variable, `$GLOBUS_LOCATION` points to the UniCluster Express installation directory and is used by Globus Toolkit components. Where required, you may set this environmental variable, or enter the complete path to the UniCluster Express installation location.

Submitting and Managing Jobs

Using qsub

To submit a job, run:

```
[user@host ~]$ qsub <install-dir> /sge/examples/jobs/simple.sh
Your job 1 ("simple.sh") has been submitted.
```

where **qsub** is the Grid Engine submit command and `install-dir/sge/examples/jobs/simple.sh/filename` is the path to a shell script containing the job parameters and executables.

For additional details on using **qsub**, run:

```
[user@host ~]$ qsub -help
```

When the job successfully completes, the user finds two files from this job:

```
[user@host ~]$ ll
total 4
-rw-r--r--  1 user user  0 Oct  3 13:45 simple.sh.e1
-rw-r--r--  1 user user 58 Oct  3 13:45 simple.sh.o1
```

The job script, `simple.sh`, is a simple shell script which prints the date and time. Requirements for a job are that the job must be a shell script and that it may call a binary executable, which must be valid for the machine on which it runs. For example, the `simple.sh` file contains:

```
# This is a simple example of a Grid Engine batch script
# request Bourne shell as shell for job
#$ -S /bin/sh
## print date and time
date
# Sleep for 20 seconds
sleep 20
# print date and time again
date
```

In this version of UniCluster Express, by default the main node is not part of the default `all.q` queue. As a result, to submit jobs on the main node (for example in a single-node cluster environment) you must use **qsub -now -y** to submit the job immediately. This is required unless your System Administrator has added the main node to `all.q`; contact your System Administrator for additional details.

Using qstat

To check the status of a job after you have submitted it, run:

```
[user@host ~]$ qstat
job-ID prior name      user  state submit/start at    queue
slots ja-task-ID
-----
-----
1  0.00000 simple.sh  user   qw   10/03/2007 13:45:14    1
```

For full information on **qstat**, run:

```
[user@host ~]$ qstat -help
```

Using qdel

To delete a job, run:

```
user@host ~]$ qdel 1
```

where `qdel` is the Grid Engine delete job command and 1 is the job ID issued when the job is accepted.

For additional details on using **qdel**, run:
`[user@host ~]$ qdel -help`

Finding Queues and Hosts

Using **qconf**

Another Grid Engine command you may find useful is **qconf**; this command displays information about queues and execution and submit hosts.

To display queue information, run:

```
[user@host ~]$ qconf -sq { cluster-queue | queue-instance |  
queue-domain }
```

where `cluster-queue`, `queue-instance`, and `queue-domain` are replaced with system specific information.

To display a list of submit hosts, use:

```
[user@host ~]$ qconf -ss
```

To display a list of the ACLs (Access Control List):

```
[user@host ~]$ qconf -sul
```

To list entries in one or more ACLs use:

```
[user@host ~]$ qconf -su acl-name[,...]
```

Grid Engine ACLs consist of user account names and UNIX group names, with the UNIX group names identified by a prefixed `@` symbol. Using these commands allows you to determine to which ACLs your account belongs. For complete details on using **qconf**, run:

```
[user@host ~]$ qconf -help
```


Chapter 3: Using Ganglia

This chapter gives information on using Ganglia and is designed to help first-time Ganglia users; for complete Ganglia documentation, please refer to the Ganglia Website [\[http://www.ganglia.info/\]](http://www.ganglia.info/).

Ganglia is a scalable distributed monitoring system for high-performance computing that allows the user to remotely view live or historical statistics for all machines being monitored. Ganglia may be used from any UniCluster Express workstation or login node.

Throughout this chapter, *install-dir* refers to the UniCluster Express installation location. The environmental variable, *\$GLOBUS_LOCATION* points to the UniCluster Express installation directory and is used by Globus Toolkit components. Where required, you may set this environmental variable, or enter the complete path to the UniCluster Express installation location.

Using Ganglia Command Line Tools

There are two command line tools that work with **gmond** to add custom metrics and query the current state of a cluster.

1. **gmetric** allows you to easily monitor any arbitrary host metrics;
2. **gstat** allows you to get status report for the cluster.

Using gmetric

gmetric allows you to monitor any arbitrary host metrics by expanding on the core metrics. All Ganglia metrics have a name, value, type and optional units.

For example, assuming a program called *cputemp* exists on a monitored machine, you could check the CPU temperature by using:

```
[user@host ~]$ gmetric --conf=install-dir/etc/gmond.conf --  
nametemp --value `cputemp` --type int16 --units Celcius
```

Table 2, “[Options for gmetric](#)” gives the available options for **gmetric**.

Table 2. Options for gmetric

Option	Description
<code>-h --help</code>	Displays help.
<code>-V --version</code>	Displays the tool version.
<code>-c --conf=STRING</code>	Specifies the configuration file to use for finding send channels; this file is located at <code>\$GLOBUS_LOCATION/etc/gmond.conf</code> .
<code>-n --name=STRING</code>	Specifies the name of the metric.
<code>-v --value=STRING</code>	Specifies the value of the metric.
<code>-t --type=STRING</code>	Specifies the type; must be one of the following: • string • int8 • uint8 • int16 • uint16 • int32 • uint32 • float • double
	Specifies the unit of measure for the value.
<code>-s --slope= v</code>	Specifies the slope and must be one of the following: • zero • positive • negative • both Default is both.
<code>-x --tmax=INT</code>	Specifies maximum time, in seconds, between gmetriccalls; default is 60.
<code>-d --dmax=INT</code>	Specifies the lifetime, in seconds, of the metric; default is 0, indicating no expiration.

Using gstat

gstat is a utility that gets the status report for the cluster. **gstat** connects with the gmond process and outputs a load-balanced list of cluster hosts. For example:

```
[user@host ~]$ gstat -a
```

Table 3, “[Options for gstat](#)” gives valid options for use with **gstat**.

Table 3. Options for gstat

Option	Description
<i>-h --help</i>	Displays help.
<i>-V --version</i>	Displays the tool version.
<i>-a --all</i>	Lists all hosts; default is off.
<i>-d --dead</i>	Prints only the hosts which are dead; default is off.
<i>-m --mpifile</i>	Prints a load-balanced <i>mpifile</i> ; default is off.
<i>-1 --single_line</i>	Prints host and information all on one line; default is off.
<i>-l --list</i>	Prints only the host list; default is off.
<i>-iSTRING --gmond_ip STRING</i>	Specifies the IP address of the gmond host to query; default is the local host, 127.0.0.1.
<i>-pINT --gmond_port INT</i>	Specifies the gmond port to query; default is 8649.

Chapter 4: Using the Monitor Console

This chapter gives information on using the Monitor Console, a GUI tool for monitoring jobs on the cluster. Information on installing this component is in the "UniCluster Express 3.2 Installation Guide", and configuration details are in the "UniCluster Express 3.2 Administrator's Guide."

Throughout this chapter, *install-dir* refers to the UniCluster Express installation location. The environmental variable, *\$GLOBUS_LOCATION* points to the UniCluster Express installation directory and is used by Globus Toolkit components. Where required, you may set this environmental variable, or enter the complete path to the UniCluster Express installation location.

Starting the Monitor Console

Using a Cluster Workstation

Monitor Console is installed by default on UniCluster Express workstation nodes. The Monitor Console is started by:

1. Configuring the shell environment by sourcing the appropriate environment setup script:
 - bash or sh users execute:
`[user@host ~]$ . $GLOBUS_LOCATION/unicluster-user-env.sh`
 - csh or tcsh users execute:
`[user@host ~]$ source $GLOBUS_LOCATION/unicluster-user-env.csh`

where *\$GLOBUS_LOCATION* is the path to the UniCluster Express installation directory.

NOTE	UniCluster Express does not support the use of zsh shell.
-------------	---

2. Logging into the cluster by running:
`[user@host ~]$ unicluster-grid-logon`
3. Running:
`[user@host ~]$ unicluster-mgmt-console`

Details on options for `unicluster-mgmt-console` are in the Table 4, “[Options for unicluster-mgmt-console](#)”.

Syntax for **`unicluster-mgmt-console`** is:
`unicluster-mgmt-console -option ..`

For example:

```
[user@host ~]$ source &lt;install-dir&gt;/unicluster-user-env.sh
[user@host ~]$ unicluster-grid-logon
[user@host ~]$ unicluster-mgmt-console
```

Accepted options for **`unicluster-mgmt-console`** are in the table below:

Table 4. Options for unicluster-mgmt-console

Option	Description
<code>-a --anonymous</code>	Use anonymous authentication; requires either <code>-m conv</code> transport (https) security.
<code>-d --debug</code>	Enables the debug mode.
<code>-h --help</code>	Displays help.
<code>-c --serverCertificate</code>	Specifies a file with server's certificate used for encryption; used in the case of GSI Secure Message encryption.
<code>-e --eprFile file</code>	Specifies an EPR from the designated for file for use with the service.
<code>-f --descriptor file</code>	Sets client security descriptor; overrides all other security settings.
<code>-g --delegation mode</code>	Performs delegation; accepted values are <code>limited</code> or <code>full</code> . Requires use of <code>-m conv</code> .
<code>-k --key name value</code>	Specifies the resource key.
<code>-l --contextLifetime value</code>	Specifies lifetime of context created for GSI Secure Conversation; requires use of <code>-m conv</code> .
<code>-m --securityMech type</code>	Sets the authentication mechanism; use <code>msg</code> for GSI Secure Message, or <code>conv</code> for GSI Secure Conversation.
<code>-p --protection type</code>	Sets the protection level; accepted values are <code>sig</code> for signature or <code>enc</code> for encryption.
<code>-s --service URL</code>	Specifies the service URL.
<code>-z --authorization type</code>	Sets the authorization; acceptable values are <code>self</code> , <code>host</code> , <code>none</code> , or a string specifying the expected identity of the remote party.

Using a Non-Cluster Machine

The Monitor Console may be run on a Windows or Linux machine that is outside of the cluster. To run on these machines, the administrator must install the tool or supply you with an appropriately configured installation file. In addition, security must be correctly configured before the Monitor Console can be used. Contact your system administrator for details.

Once Monitor Console is installed and security is configured, the Monitor Console is started on a Windows system by clicking *Start > All Programs > Univa > Monitor Console*.

On Linux machines, start the Monitor Console by running:
`./path/to/mgmt-console/unicluster-mgmt-console`

where `/path/to/` is the Monitor Console installation path on your machine.

Running the Monitor Console on a Non-Cluster Machine

Once the Monitor Console is running, you must point the monitor at the correct main node and generate a proxy from your user credentials to connect with the main node for information.

First, set the main node location by clicking on *File > Server Address*; enter the complete hostname of the cluster's main node and click **OK**.

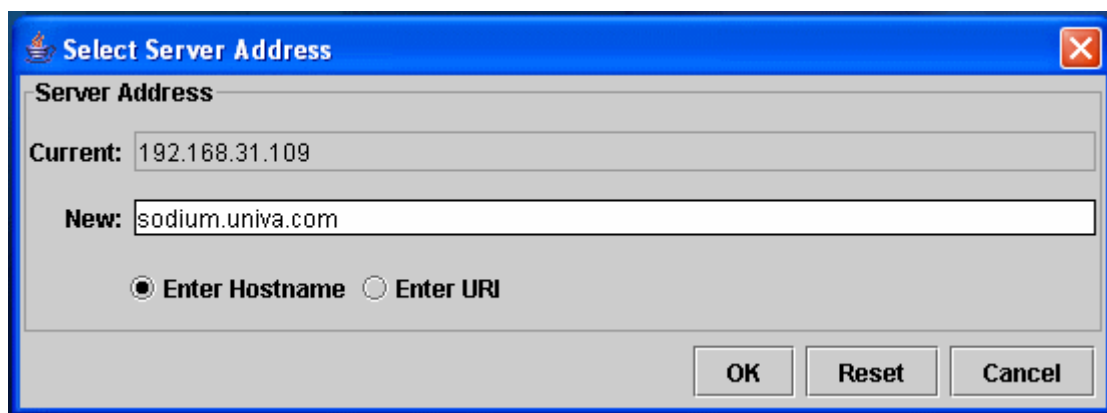


Figure 1. Entering the Main Node Address

Enter the host name of the cluster's main node. Then generate the required user credentials and install those on the Monitor Console node. On a workstation node in the cluster, create a directory called `.globus` in your home directory.

Create a set of proxy certificate and key files by running `unicluster-grid-logon` with the **-split** option. For example:

```
[user@host ~]$ unicluster-grid-logon -split -cert  
~/.globus/usercert.pem  
-key ~/.globus/userkey.pem
```

This command generates the required *usercert.pem* and *userkey.pem* files in your *.globus* directory. These files are valid for 12 hours by default; to generate files valid for a longer duration, use the **-t hours** option. Copy the *.globus* directory containing the *usercert.pem* and *userkey.pem* files to the non-cluster machine.

On a Windows machine, copy the *.globus* directory to `C:\Documents and Settings\username\`.

On Linux machines, copy the directory to your home directory.

The actual file location may differ from that given; however, if you change the directory location, you must specify the new location in Monitor Console by choosing *File > Manage Proxy* and click **Options**.

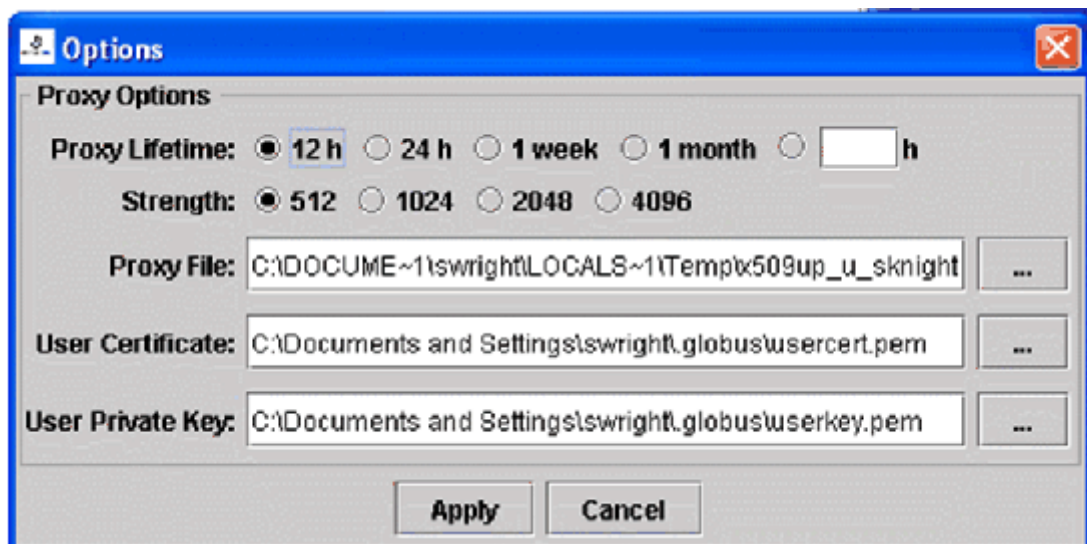


Figure 2. Entering a New Credential Location

Enter the new location of the user credential. In the **Options** dialog box, set the proxy file, user certificate, and user private key file locations, then click **Apply**. For security purposes, the location you choose for these files should not be readable by other users.

Once *usercert.pem* and *userkey.pem* files are copied to the machine on which you are working, select *File > Manage Proxy > Enter a password*, and click **Create** to generate the required proxy credentials.

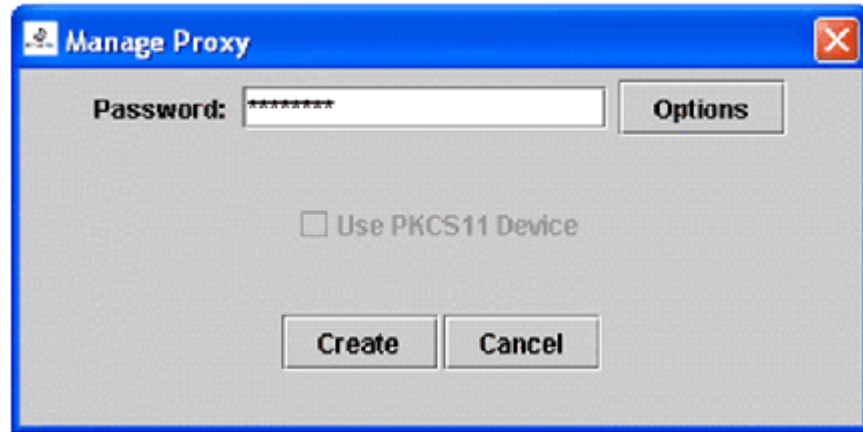


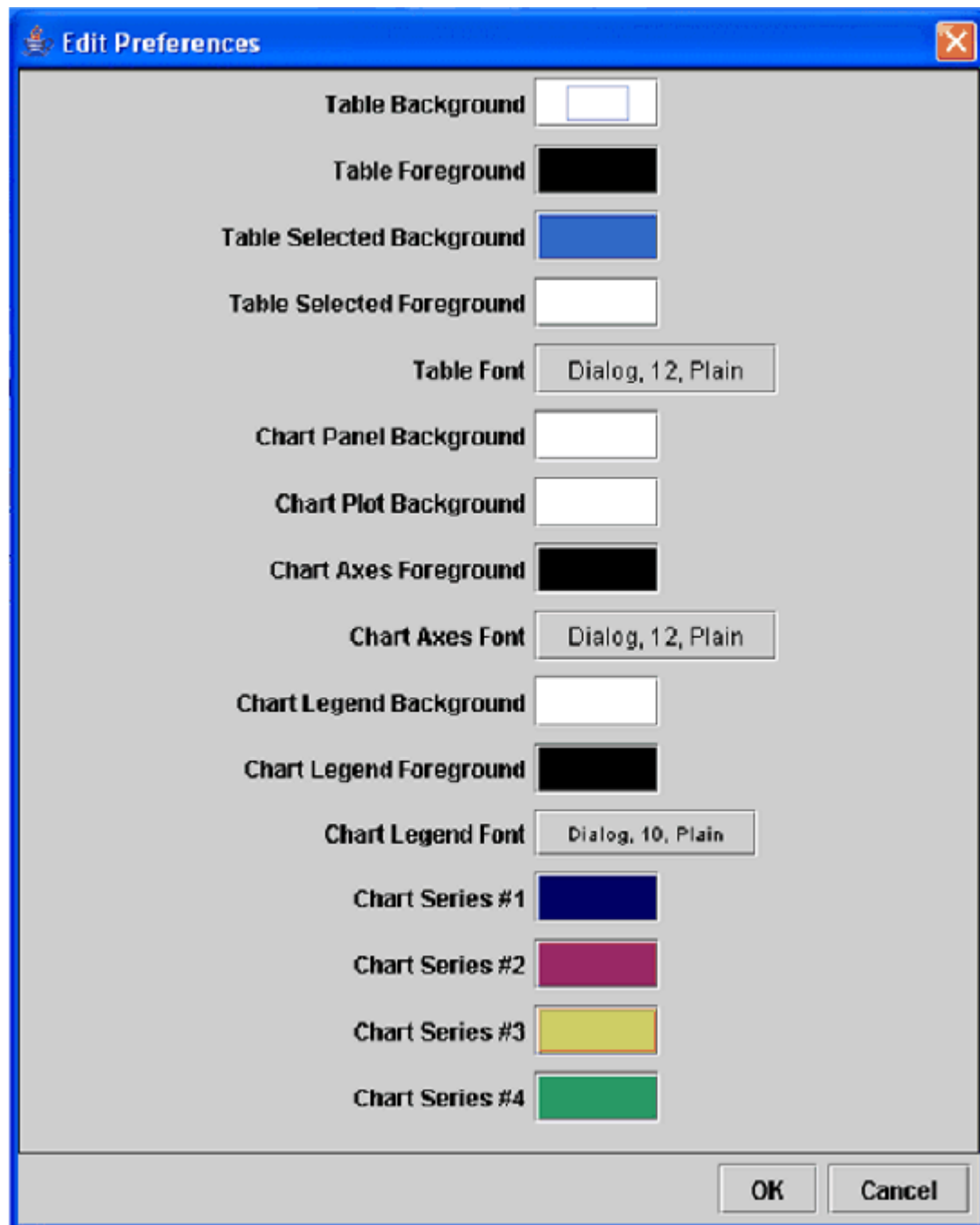
Figure 3. Creating a Proxy Credential

Enter a password in the *Manage Proxy* dialog box and click **Create**.

NOTE	The <code>unicluster-grid-logon</code> command generates credentials which are valid for 12 hours; to generate a credential valid for a longer period of time, use the <code>-t</code> option with the number of hours (maximum of 168) for the credential lifetime..
-------------	---

Changing Monitor Console Preferences

You can change the colors and textual display used in the Monitor Console easily by clicking *Edit > Preferences*.



Clicking on one of the color squares in the *Edit Preferences* dialog box opens a color picker with all available colors for the visual display. To change the display color for that particular item, choose a color by clicking on it in the color picker, then click **OK**.

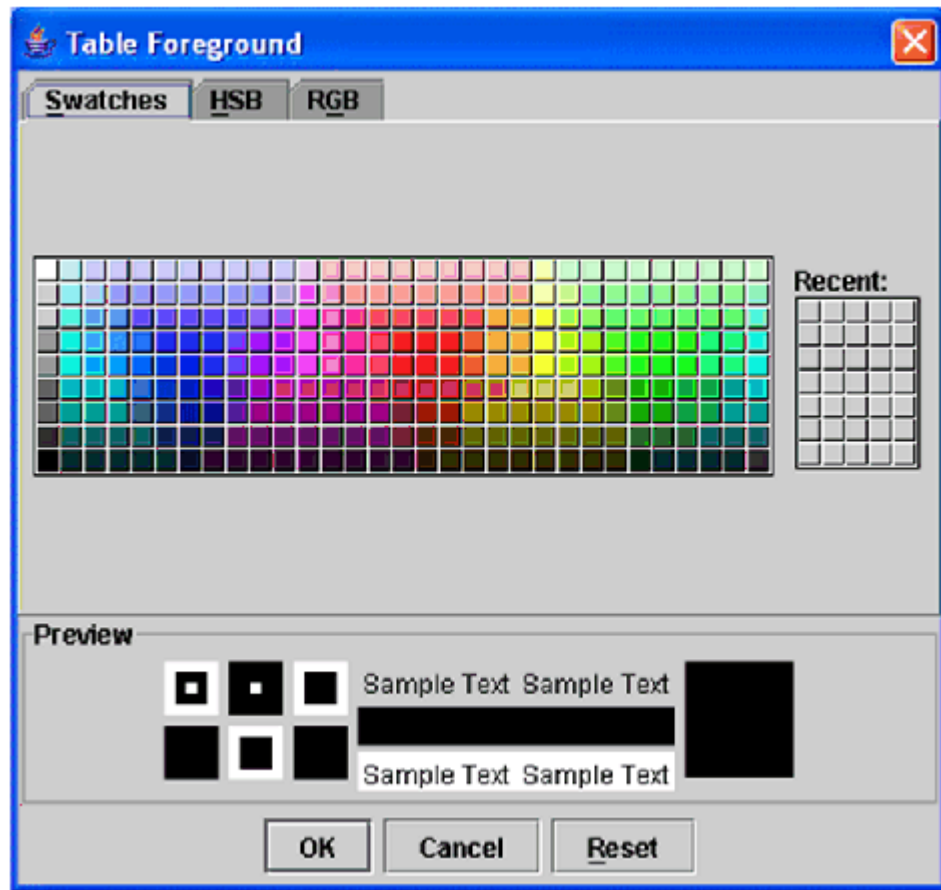


Figure 4. Changing Display Colors

Likewise, you can choose a different font for textual display of tables, and charts by clicking on the font name in the *Edit Preferences* dialog; in the *Chart Legend Font* dialog box that opens, choose the font, style, and size, then click **OK**.

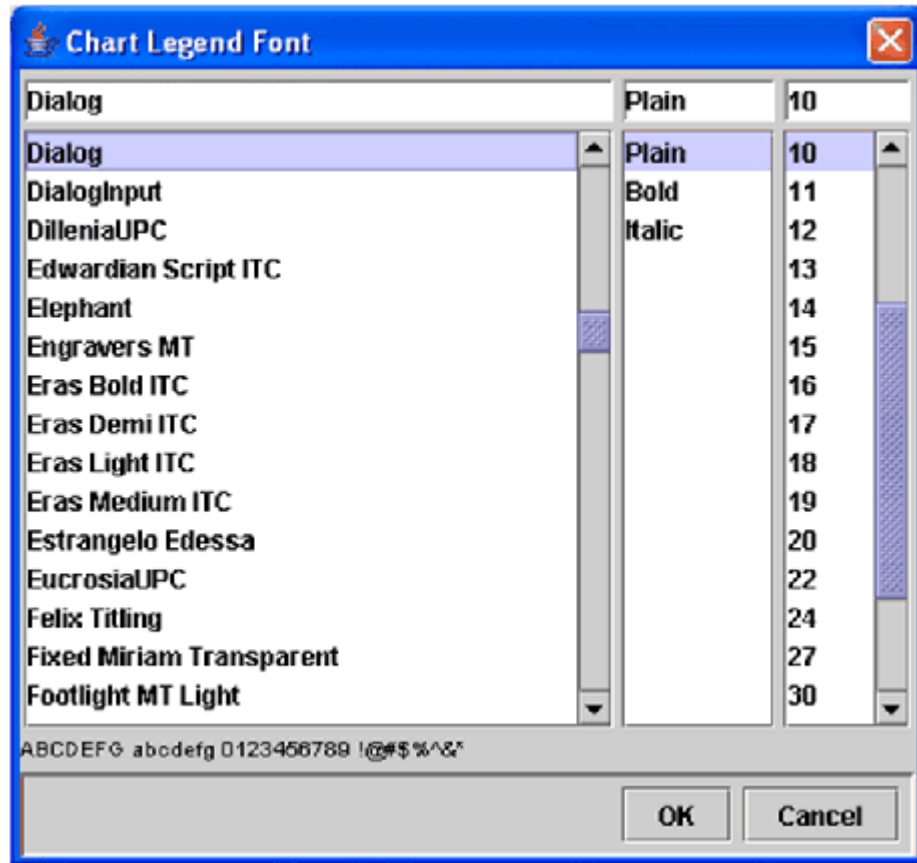


Figure 5. Changing Display Fonts

Changing Displayed Time Ranges

In the upper right corner, from the *Time Range* menu you can choose a specific time period for data review. The default is for the last hour; other choices are **Last Day**, **Last Week**, or a custom range, in which you can define a specific time period for viewing.

You can set color and text preferences for the Monitor Console display.

The image below displays the dialog box that allows you to specify a time range for data display.

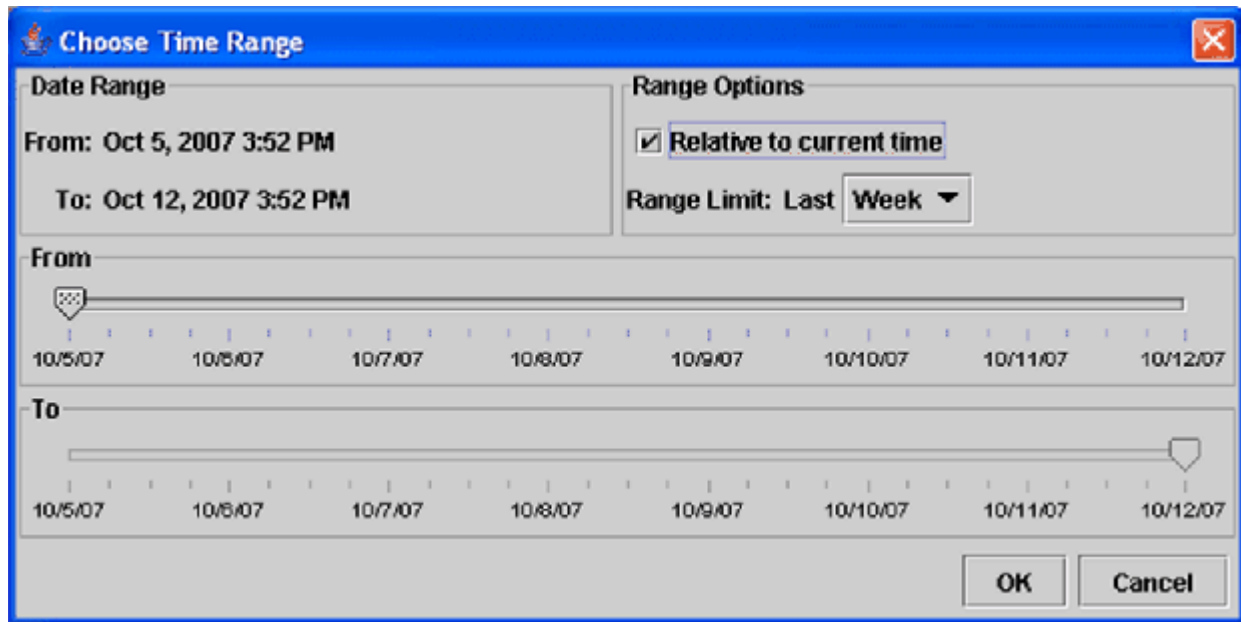


Figure 6. Choose Time Range

In the **Choose Time Range** dialog box, move the sliders to the appropriate settings to set the time range for the display.

Using the Cluster Summary Tab

The **Cluster Summary** tab shows the overall status of the cluster nodes. In the upper left, the Cluster Health pane gives a summary of the cluster as a whole, showing the percentage of machines that are running and down.

NOTE	Depending upon how your administrator installed and configured the cluster, the Monitor Console may detect and display <i>all</i> the machines on your network running a Ganglia gmond daemon, not just UniCluster Express nodes. Contact your administrator for details on the Monitor Console setup.
-------------	---

The panes immediately under **Cluster Health**, the **CPU** and the **Network** panes, show the percentage of CPU being used and the byte throughput of the network, respectively. In both of these graphs, you may click and drag over an area to see more specific details.

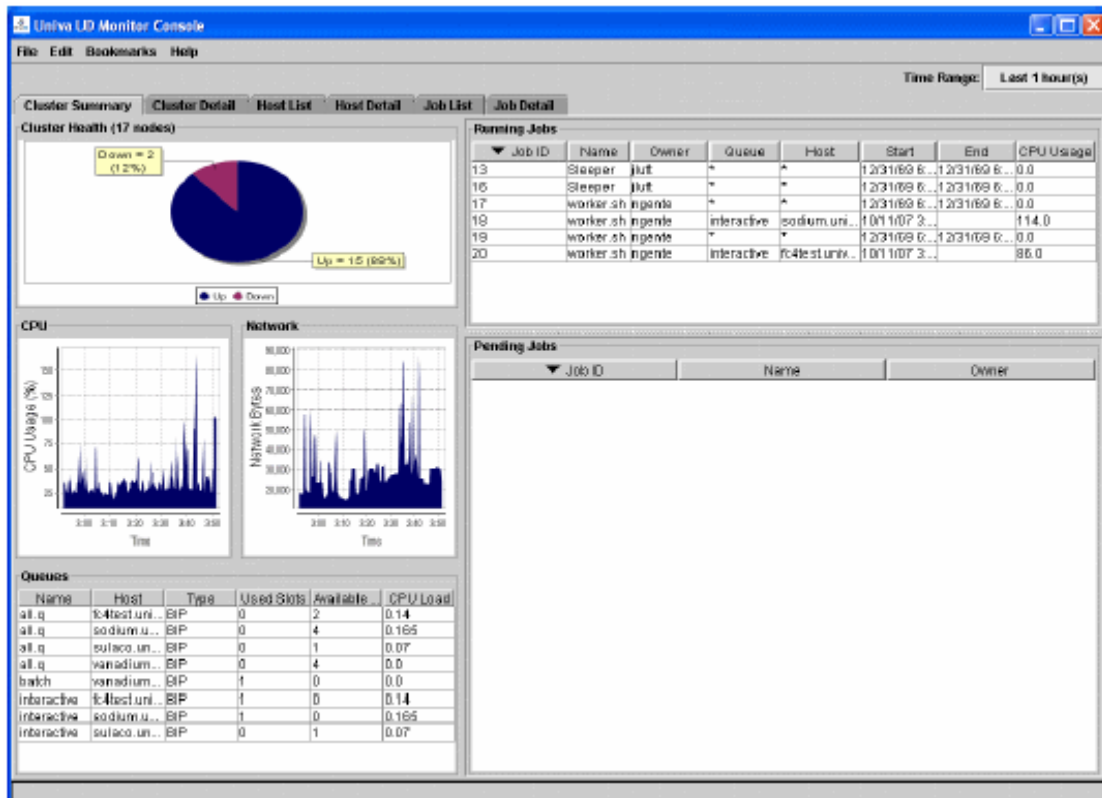


Figure 7. The Cluster Summary Tab

On the bottom left, the **Queues** pane gives a list of the available queues in the cluster, with information on which nodes they are running on, their types, current usage, and CPU load. Double click on one of the queue names opens the **Job List** tab, detailed in the section [Using the Job List Tab](#).

Below the Time Range menu is the Running Jobs pane; this pane lists running and completed Grid Engine jobs, along with their job ID and owner. Running jobs will also show the queue and hostname, as well as the start time; completed jobs show only the start and end times. Double click on one of the jobs brings up the job details in the "Job Detail" tab, detailed in the section [The Job Detail Tab](#).

Directly below Running Jobs is the Pending Jobs pane, in which all pending Grid Engine jobs are listed, along with their job ID and owner. Double click on one of the pending job names loads the details of that job in the "Job Details" tab, detailed in the section [The Job Detail Tab](#).

Using the Cluster Detail Tab

The Cluster Detail tab shows more in-depth information about status of nodes on the cluster. As with other Monitor Console graphs, click and drag on part of one of the graphs to load more detailed information; right click the detailed graph allows you to apply that time range to all graphs on the tab.

The graphs on the Cluster Detail tab are (from top left to bottom right):

CPU Load – shows the load average over time;

Memory Usage – shows the cached, buffered, shared, and free memory over time;

CPU Usage – shows the percentage of CPU use over time;

Network Usage – shows the number of packets in and out over time;

Process Count – shows the number of running processes as well as the total number of processes;

Free Disk – shows the free and used gigabytes over time.

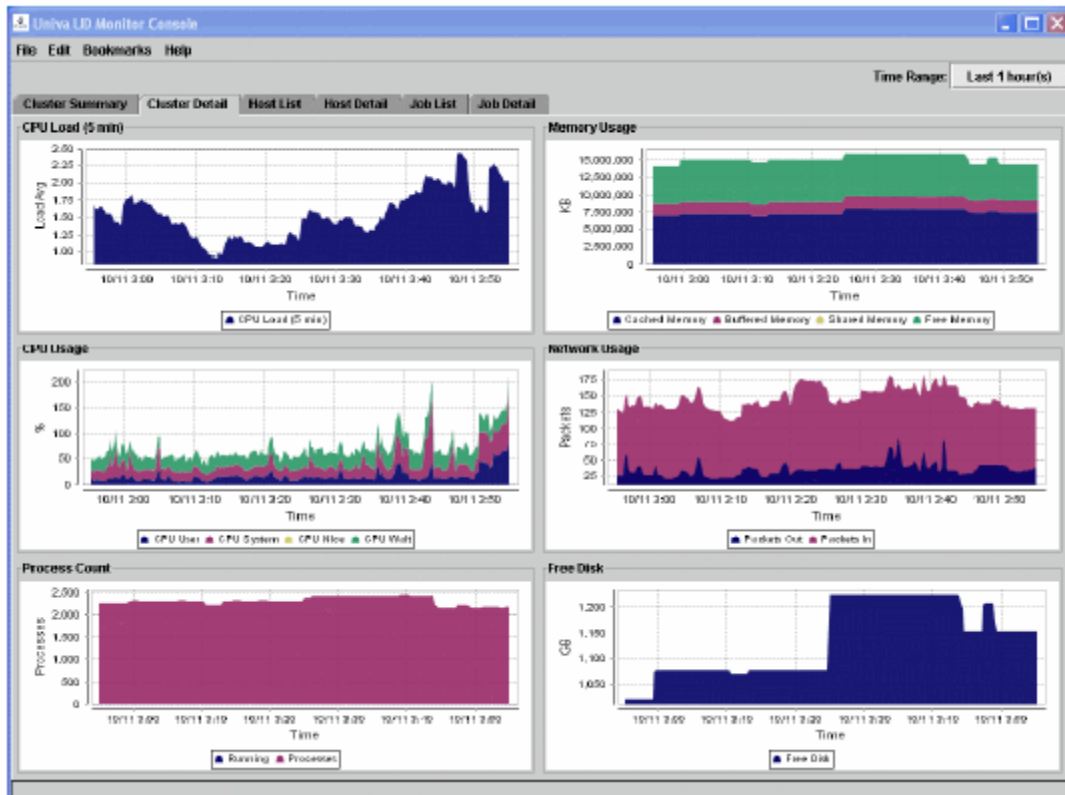


Figure 8. The Cluster Detail Tab

Each of these graphs has a right click menu with menu choices:

- The CPU Load graph right click allows you to view hosts by their CPU load.
- The Memory Usage graph right click menu offers the choice to view hosts by cached memory use, buffered memory use, shared memory use, or free memory.
- The CPU Usage right click menu allows you to view hosts by CPU user, CPU system, CPU nice, or CPU wait time.
- The Network Usage right click menu gives the choice to view hosts by the number of packets in or number of packets out.
- The Process Count right click menu offers the choice to view hosts by either the running processes or total processes.
- The Free Disk right click menu allows you to view hosts by free disk space.

Selecting any of these right click options on the graphs loads the appropriate details in the **Host List** tab, detailed in the section [Using the Host List Tab](#).

Using the Host List Tab

The **Host List** tab shows general information about nodes on the cluster and their status. As with other Monitor Console graphs, click and drag on part of one of the graphs will load more detailed information. Double click on any hostname to open the **Host Detail** tab, with more detailed information about that specific host; details on the **Host Detail** tab are in the section [Using the Host Detail Tab](#)

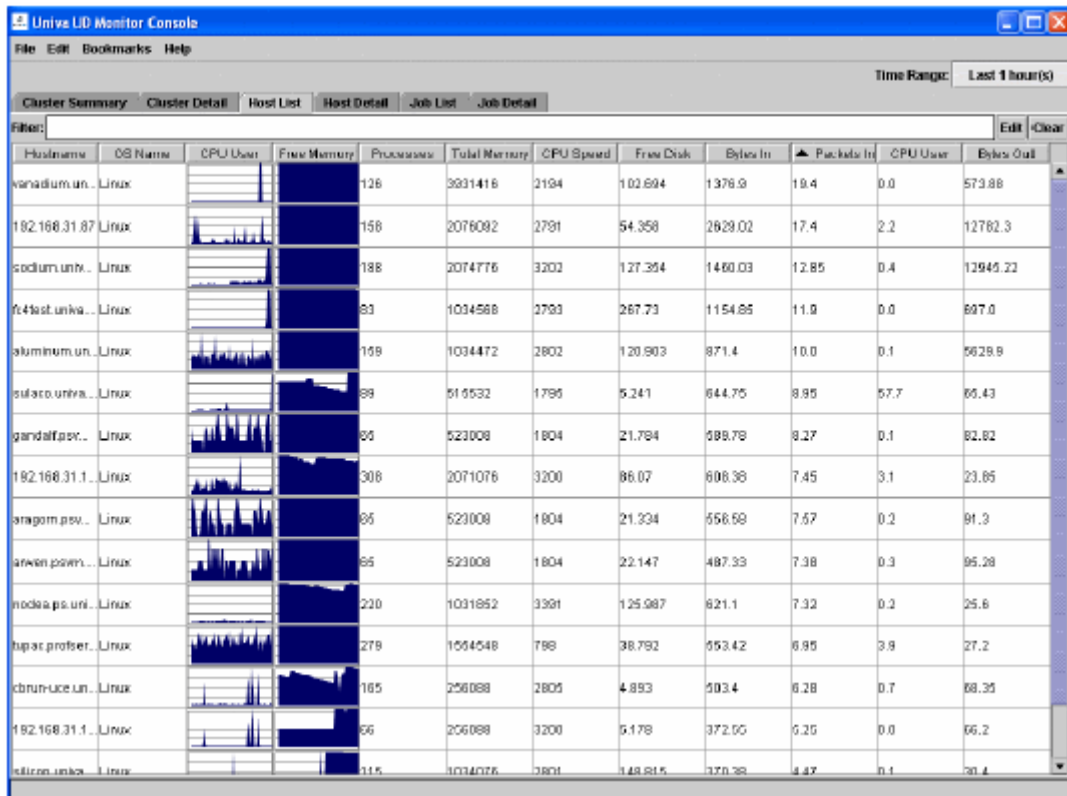


Figure 9. The Host List Tab

Each host entry in the **Host List** tab lists information on the host; the following fields are displayed:

Hostname – Gives the node hostname.

OS Name – Lists the node's operating system.

CPU User – Gives the historical CPU usage, in a graphical format, while executing at the user level; usage is given as a percentage.

Free Memory – Gives the amount of available memory for the node.

Processes – Gives the total number of processes on the node.

Total Memory – Gives the total amount of available memory on the node.

CPU Speed – Gives the speed, in MHz, of the node's CPU.

Free Disk – Gives the total free disk space for that node.

Bytes In – Gives the number of bytes in per second on that node.

Packets In – Gives the number of packets into the node per second.

Last Update Time – Lists the last time the node sent updated information to the Ganglia **gmetad** daemon.

CPU Load (5 min.) – Gives the five minute load average for the node.

CPU User – Gives the current numerical CPU usage, while executing at the user level; usage is given as a percentage.

Packets Out – Gives the number of packets out of the node per second.

Bytes Out – Gives the number of bytes out per second.

Right click on any of the column names on the **Host List** tab and a menu is displayed that allows you to configure the columns to be displayed and how they are to be displayed.

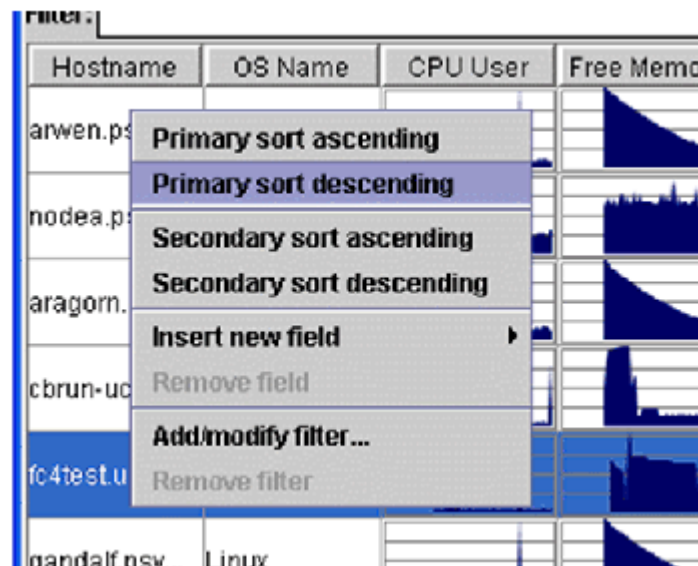


Figure 10. Right Click Menu for the Host List tab

For example, right click on the *Hostname* column; select **Primary sort ascending**. The column reorders to show the hostnames in ascending alphabetical order. You can also choose to display hostnames by descending order, and there is an option to add a secondary sort as well.

You can add or delete columns displayed to suit your specific needs.

For example, right click the numerical *CPU User* column and select **Remove Field**; this field is removed from the display. Now, right click one of the headings and choose **Insert New Field**. A sub menu appears with several categories of field types; choose **CPU**, then **CPU Idle**. A new column **CPU Idle** is shown; this field gives the idle time of the CPU for that node.

The [Available Fields for the Host List Tab](#) displays a list of available metrics that can be displayed in the Monitor Console.

Table 5. Available Fields for the Host List Tab

Field Category	Field	Description
System	Number of CPUs	Lists the number of CPUs for that node.
	Machine Type	Gives the node architecture.
	OS Name	Gives the operating system for that node.
	OS Release	Gives the kernel information for that node.
	Location	In a multi-cluster installation, the Location identifies the cluster the node is associated with.
Memory	Boot Time	Gives the current up time for that node since the last boot.
	Total Swap	Gives the total amount of swap memory for that node.
	Free Memory	Gives the total amount of free memory for that node.
	Shared Memory	Gives the total amount of shared memory for that node.
	Buffered Memory	Gives the total amount of buffered memory for that node.
Storage	Cached Memory	Gives the total amount of cached memory for that node.
	Free Swap	Gives the total amount of free swap memory for that node.
	Total Disk	Gives the total amount of disk space on that node.
	Partition Max Used	Gives the total amount of a partition used on that node.
	Free Disk	Gives the total free disk space on that node.
Load	CPU Load (1 min)	Gives the average CPU load on that node for a one minute period.
	CPU Load (5 min)	Gives the average CPU load on that node for a five minute period.
	CPU Load (15 min)	Gives the average CPU load on that node for a fifteen minute period.

Table 5. Available Fields for the Host List Tab

Field Category	Field	Description
CPU	CPU User	Gives the CPU use as a percentage, while the node is executing at the user level.
	CPU System	Gives the total CPU usage across that node.
	CPU Nice	Gives the percentage of CPU utilization while executing at the user level with nice priority.
	CPU Abs Idle	Gives a percentage of the absolute CPU idle time.
	CPU Wait	Percentage of time that the CPU or CPUs were idle and the system did not have an outstanding disk I/O request.
	Processes	Gives the number of currently running processes on that node.
	Running	Gives the number of running jobs on that node.
Input/Output	Bytes In	Gives the total number of bytes in per second.
	Bytes Out	Gives the total number of bytes out per second.
	Packets In	Gives the total number of packets in per second.
	Packets Out	Gives the total number of packets out per second.

Almost all of the metrics have both a graphical display and a numerical display of the same name, allowing you to choose the format that best suits your needs. Graphical columns are noted by a small red graph next to the column name; numerical columns are distinguished by the white square containing "0.0."

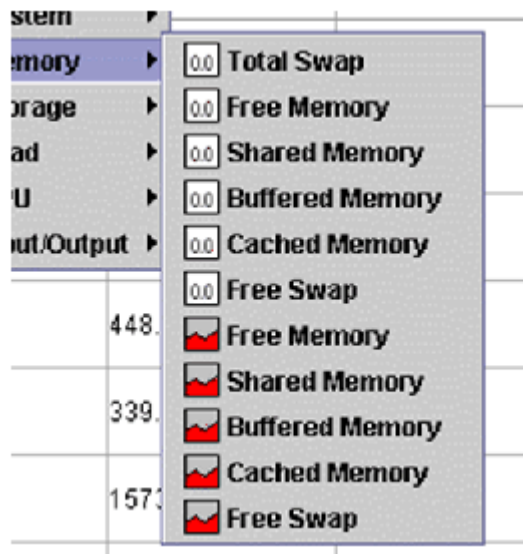


Figure 11. Graphical and Numerical Columns

The small red graph to the left of the column name indicates the column is a graphical display; the small white square containing "0.0" indicates a numerical display.

Filtering Information

Just above the list of hosts, the **Filter** text box allows you to filter the host list information. By default, there is no filter; to add a filter, click **Edit** to the right of the **Filter** text box and enter a filter; when done, click Done. To add a filter expression, right click a column and select **Add/Modify Filter**.

Filters may be written based on any combination of columns. For each column, an expression is entered to define whether a row with a particular column value should be listed or not.

For example:

```
(mem_total > "800000") && (proc_total < "150")
```

This filter lists all hosts that have over 800000 MB in memory and less than 150 total processes.

Filter expressions are written using the Ganglia field name; to determine a field name, hold the mouse over the column heading. The default column layout includes the following field names:

- The Hostname column field name is *name*.
- The CPU User column field name is *cpu_user*.
- The Processes column field name is *proc_total*.
- The Total Memory column field name is *mem_total*.
- The CPU Idle column field name is *cpu_idle*.
- The CPU Speed column field name is *cpu_speed*.
- The Bytes In column field name is *bytes_in*.
- The Packets In column field name is *pkts_in*.
- The Last Update Time column field name is *last_update*.
- The CPU Load (5 min.) column field name is *load_five*.
- The Packets Out column field name is *pkts_out*.
- The Bytes Out column field name is *bytes_out*.

Filter expressions may include the following operators:

- Equality (==)
- Inequality (!=)
- Less-than (<)
- Less-than-or-equal (<=)
- Greater-than (>)
- Greater-than-or-equal (>=)
- Regular expression (regexp())
- Logical AND (&&)
- Logical OR (||)
- Logical XOR (^)

Saving and Organizing Bookmarks

Once you have cluster information displayed to suit your needs, you can bookmark, or save, the view, by choosing *Bookmarks > Bookmark* this view. In the **Save Bookmark** dialog box, enter a name for the bookmark, then click **OK**.

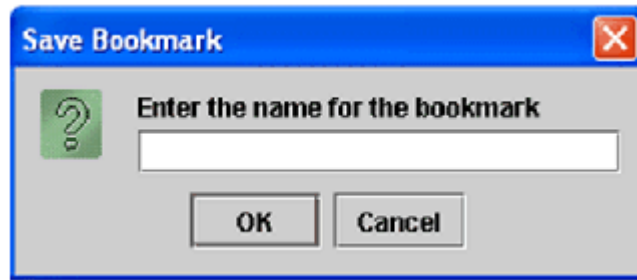


Figure 12. Saving a Bookmark

Once you have built a view that suits your needs, save it using the **Bookmark** feature.

You can organize saved bookmarks by choosing *Bookmarks > Organize Bookmarks* from the menu. Right click on one of the saved bookmarks and it allows you to choose to either clone or delete the selected bookmark.

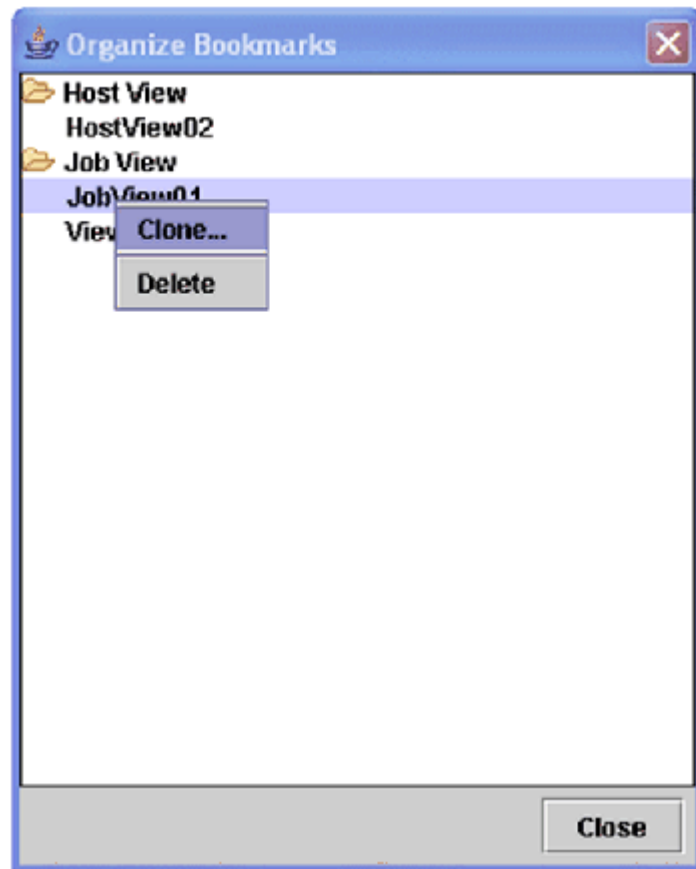


Figure 13. Organizing Bookmarks

Selecting the **Organize Bookmarks** option allows you to clone or delete saved bookmarks.

Using the Host Detail Tab

When you double click on any hostname in the **Host List** tab, it displays the **Host Detail** tab, showing additional details about that host. On the left of the **Host Detail** tab, the **Detailed Metrics** list displays all the available metrics for that host and the current value.

In the center of the tab, the **Historical** pane allows you to view details about the node's historical data in a graphical format; by default, **CPU Usage** and **CPU Load** are shown. You can toggle the graphical displays of any of the metrics listed by checking and unchecking the box next to the metric name.

You can change the time range displayed on graphs on this tab by clicking and dragging across a specific portion of the graph. To have this time period displayed on all graphs displayed, right-click the revised graph and select Apply Time Range.

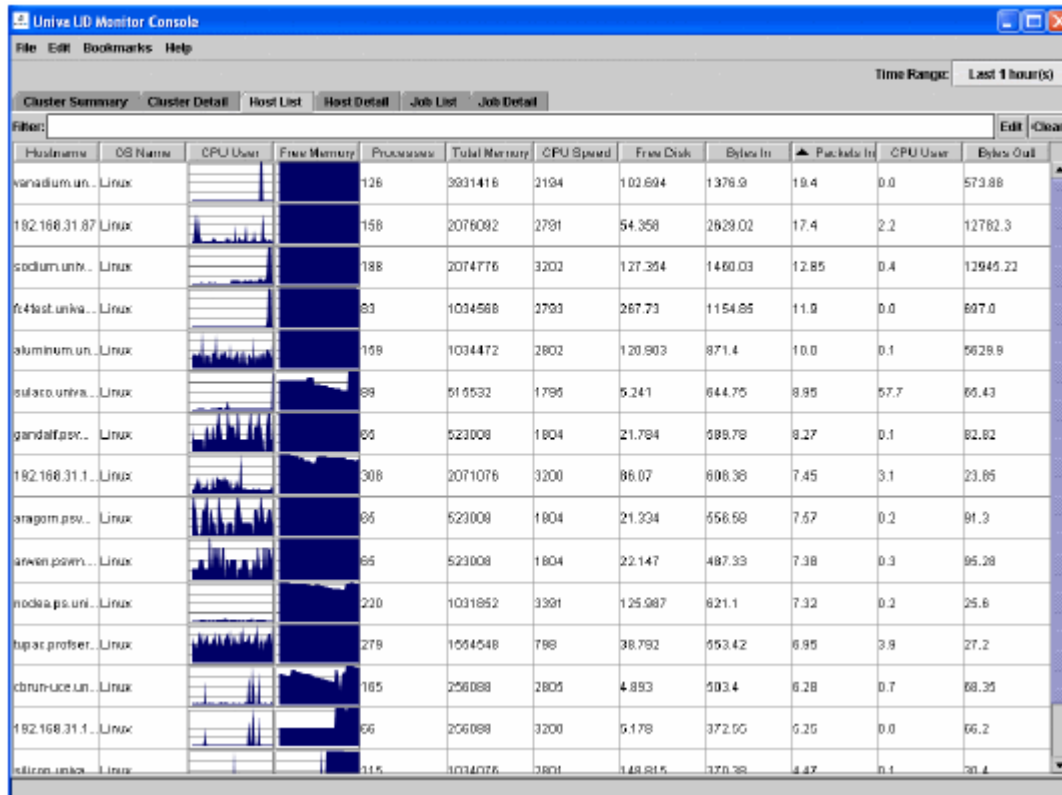


Figure 14. The Host Detail Tab

To the right of the Historical pane is the Queues and Jobs panes.

The Queues pane shows the active queues on the cluster, which host the queue runs on, type of queue, used slots, available slots, and CPU Load.

When you double click on a queue name, the Job List tab opens with details about jobs that have run, or are currently running, on that queue. Details about the Job List tab are in [Using the Job List Tab](#).

The Jobs pane gives information about completed and running jobs. Information in this pane includes the job ID, name, owner, queue, host, start and finish times for completed jobs, and CPU usage.

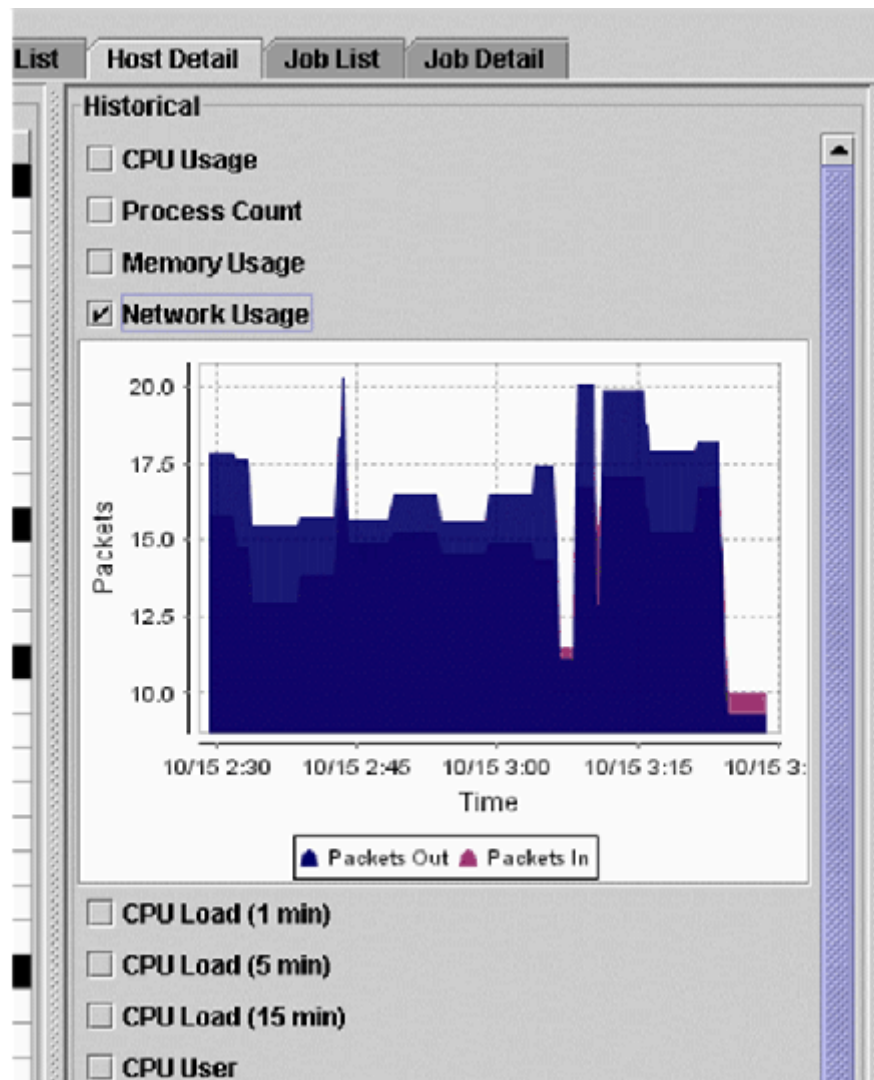


Figure 15. Displaying Graphs for Host Metrics

You can display or hide any of the host metrics in the Historical pane by checking or unchecking the box next to the metric name.

When you right click on a column heading in the Jobs pane, you can sort information by ascending or descending listings in that field; as with the Host List tab right click menu, you may also apply a secondary sort if desired.

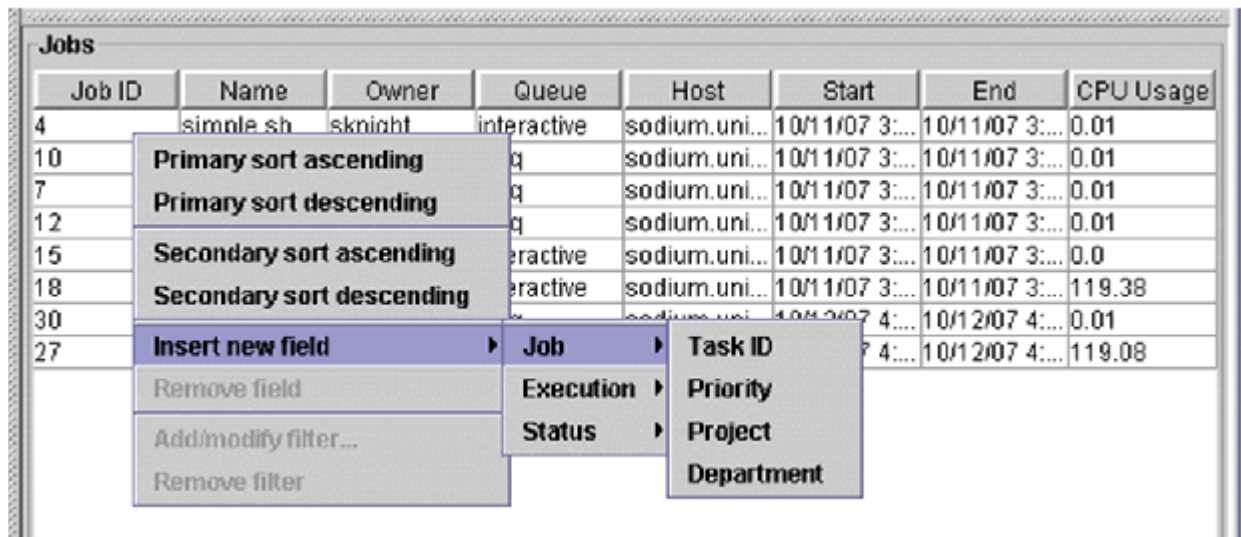


Figure 16. Adding a Column for Job Information

Right click on a column heading in the Jobs pane to let you add or remove columns, as well as reorder the results.

Right click on a column in the Jobs pane also gives you the choice of adding additional information fields or deleting fields which you do not need; the exception is the Job ID field, which cannot be removed.

[Available Fields for the Jobs Pane](#) displays a list of available metrics that can be displayed in the **Jobs** pane.

Table 6. Available Fields for the Jobs Pane

Field Category	Field	Description
Job	Task ID	For jobs that have multiple tasks, this column gives the specific Task ID for that job.
	Priority	Gives the job priority.
	Project	If defined, this column lists the project the job is identified with.
	Department	If defined, this column gives the department the job is identified with.
Execution	Status	Gives the current execution status of the job.
	Submit	Gives the job submission time.
	Elapsed	Gives the elapsed job run time.
Status	Memory Usage	Gives the total memory usage of that job.
	I/O Usage	Gives the total input/output of that job.

Double click on any job ID number to open the Job Detail tab; the Job Detail tab is detailed in the section [The Job Detail Tab](#).

Using the Job List Tab

The **Job List** tab presents information about the queues on each host and the jobs that have run, or are currently running, in those queues.

On the left side of the **Job List** pane is the **Queues** pane. Information in this pane includes the queue name, which host that queue is running on, type of queue, used slots, available slots and the current CPU load. This information is similar to that given on the **Host Detail** tab, except that you can view all queues on all hosts simultaneously.

Name	Host	Type	Used Slots	Available Slots	CPU Load
all.q	fc4testuniv...	BP	0	2	0.08
all.q	sodium.univ...	BP	0	4	0.115
all.q	solace.univ...	BP	0	1	0.55
all.q	vanadium...	BP	0	4	0.0
batch	vanadium...	BP	0	1	0.0
interactive	fc4testuniv...	BP	0	1	0.08
interactive	sodium.univ...	BP	0	1	0.115
interactive	solace.univ...	BP	1	0	0.55

Job ID	Name	Owner	Queue	Host	Start	End	CPU Usage
9	simple.sh	sknight	all.q	fc4testuniv...	10/11/07 ..	10/11/07 3...	0.02
7	simple.sh	sknight	all.q	sodium.univ...	10/11/07 ..	10/11/07 3...	0.01
10	simple.sh	sknight	all.q	sodium.univ...	10/11/07 ..	10/11/07 3...	0.01
12	simple.sh	sknight	all.q	sodium.univ...	10/11/07 ..	10/11/07 3...	0.01
5	simple.sh	sknight	all.q	vanadium...	10/11/07 ..	10/11/07 3...	0.01
8	simple.sh	sknight	all.q	vanadium...	10/11/07 ..	10/11/07 3...	0.01
11	simple.sh	sknight	all.q	vanadium...	10/11/07 ..	10/11/07 3...	0.03

Figure 17. The Job List Tab

To the right of the **Queues** pane is the **Jobs** pane. Information here includes the job ID, job name, owner, queue, host, start and end times, and CPU usage for that job.

Right click on a column heading to sort data, and add or remove columns.

Available columns are the same as those for the **Host Detail** tab's **Jobs** pane, which are listed in [The Job Detail Tab](#).

In the **Job List** tab **Jobs** pane, however, you can filter job information by a specific criteria.

By default, the filter used in the **Jobs** pane is to list all jobs in the default *all.q* queue. You can easily change the filter by revising the filter definition in the **Filter** text box directly above the **Jobs** pane.

For example, click **Edit** to the right of the **Filter** text box and enter a different queue name, then click **Done**; double click the name in the **Queues** pane to change the filter easily.

Filters may be written based on any combination of columns in the **Jobs** pane. For each column, an expression is entered to define whether a row with a particular column value must be listed or not.

For example:

```
(job_queue == "all.q") && (job_owner == "jdoe")
```

This filter lists all jobs in the *all.q* job queue that are owned by the *jdoe* user.

Filter expressions are written using the Ganglia field name; to determine a field name, hold the mouse over the column heading. The default column layout includes the following field names:

- The Job ID column field name is `job_id`.
- The Name column field name is `<filename>job_name`.
- The Owner column field name is `<filename>job_owner`.
- The Queue column field name is `<filename>job_queue`.
- The Host column field name is `<filename>job_host`.
- The Start column field name is `<filename>job_start`.
- The End column field name is `<filename>job_end`.
- The CPU User column field name is `<filename>job_cpu`.

Filter expressions may include the following operators:

- Equality (`==`)
- Inequality (`!=`)
- Less-than (`<`)
- Less-than-or-equal (`<=`)
- Greater-than (`>`)
- Greater-than-or-equal (`>=`)
- Regular expression (`regexp()`)
- Logical AND (`&&`)
- Logical OR (`||`)
- Logical XOR (`^`)

Double click on one of the job names in the **Jobs** pane; the **Job Detail** tab with more detailed information about that job is displayed. The **Job Detail** tab is discussed in the section [The Job Detail Tab](#).

As with the Host List tab, you can bookmark views on the Job List tab by selecting *Bookmarks > Bookmark this view*. Saving and organizing bookmarks is discussed in the section [Saving and Organizing Bookmarks](#).

The Job Detail Tab

The **Job Detail** pane shows additional information about one of the jobs on the cluster. Double click on a job name in the **Job List** tab to open this view.

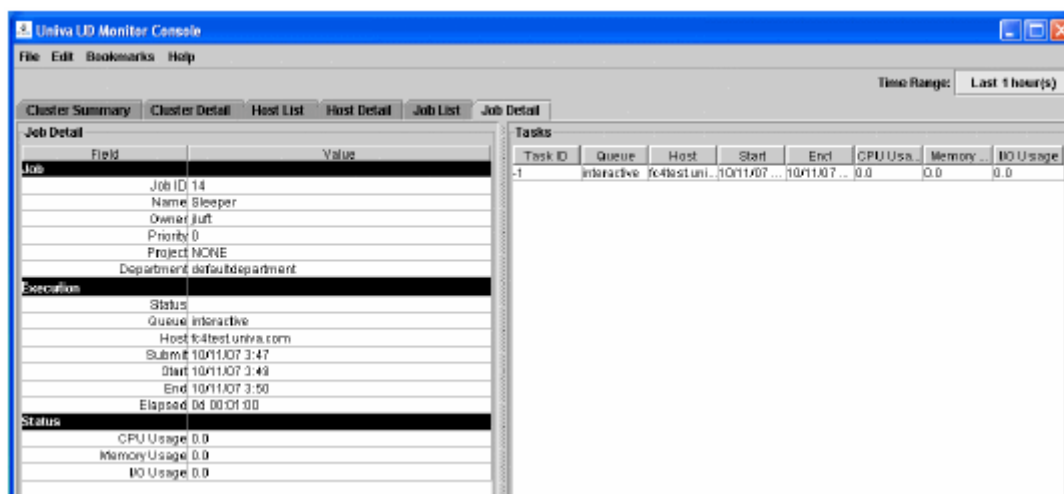


Figure 18. The Job Detail Tab

On the left is the **Job Detail** pane; this pane lists detailed information about a particular job. To the left of this pane is the **Tasks** pane, which lists all tasks associated with the job. Unless the job is an array, the **Tasks** pane will only contain one listing for the job.

Quitting the Monitor Console

To quit the Monitor Console, select *File > Quit*.

Chapter 5: Using GridFTP

This chapter provides information on using the GridFTP command line client, **globus-url-copy**, to move data. You can submit a **globus-url-copy** command on workstation and login nodes.

globus-url-copy is a scriptable command line tool that performs multi-protocol data movement of files and directories. This tool supports *gsiftp://*, *ftp://*, *http://*, *https://* and *file:///* URLs. Examples of valid URLs are:

- `gsiftp://host.domain.com:2811/data/foo.dat`
- `http://host.domain.com/mywebpage/default.html`
- `file:///foo.dat`
- `file:/foo.dat`

URLs must be in the form: `protocol://host:port/path`. If the final element is a directory, the path must end with a forward slash (/); for example:
`gsiftp://123.456.78.9:2811/home/jdoe/mydir/`

Using globus-url-copy

Syntax of **globus-url-copy** is:

```
globus-url-copy -option -option ... sourceURL destURL
```

where

- `sourceURL</filename>` is the source file path;
- `destURL</filename>` is the destination file path

For example:

```
[user@host01 ~]$ globus-url-copy
gsiftp://host01.univaud.com:2811/ home/user/ file01
gsiftp://host02.univaud.com:2811/tmp/foo
```

```
[user@host02 ~]$ ll /tmp | grep foo
-rw-r--r--  1 user user      871 Oct  3 11:29 foo<
```

Valid options for **globus-url-copy** are in the following table.

Table 7. Options for globus-urlcopy

Option	Description
-help -usage	Prints help.
-q -quiet	Suppresses all output for a successful operation.
-vb -verbose	During transfers, this option displays the number of bytes transferred, performance since the last update (currently every five seconds), and average performance for the whole transfer.
-dbg -debugftp	debugs connections and prints entire control channel protocol exchange to stderr.
-a -ascii	Converts file between ASCII and local file formats.
-b -binary	Does not apply any conversion to the files; default is on.
-f filename	Sets the file in which URL pairs for transfers are located; each line in this file contains the sourceURL and destURL. Enclose URLs with spaces in double quotes; blank lines and lines beginning with # are ignored.
-r -recurse	Copies files in subdirectories.
-notpt -no-third-party-transfers	Turns third party transfers off; default is on. NOTE Firewalls may prevent connections between servers in the case of third party transfers. This option forces globus-url-copy to relay data, doing a GET from the source and a PUT to the destination.
-rst -restart	Restarts a failed transfer.
-rst-retries retries	Specifies maximum number of times to retry before cancelling the transfer; default is 5. 0 forces infinite.
-rst-interval seconds	Specifies interval, in seconds, to wait after a failure before retrying a transfer; default is 0 (exponential backoff).
-rst-timeout seconds	Specifies maximum time, in seconds, after a failure to keep retrying; default is 0 (no timeout).

Table 7. Options for globus-urlcopy

Option	Description
-tcp-bs -tcp-buffer-size size	Specifies size, in bytes, of TCP buffer to be used by the data channel. The value chosen for the TCP buffer size limits top transfer speeds achieved and depends upon bandwidth and distance across which the data must be moved (as measured by Round Trip Time, or RTT). To calculate TCP buffer size, use the following formula: $-tcp-bs = \text{bandwidth in Mbs} * \text{RTT in ms} * 1000 / 8$ NOTE This formula assumes 'Mega' means 1000^2 rather than 1024^2, which is typical for bandwidth. For example, on a fast ethernet (100 Mbs) connection, with an RTT of 50 ms: $tcp-bs = 100 * 50 * 1000 / 8 = 625,000 \text{ bytes}$ Bandwidth value can be determined by locating the network bottleneck using system diagnostics, or the speed of the host NIC, assuming the host is the network bottleneck. To determine an RTT value, use ping or tracert. To use this value correctly, you must ping from one end of the transfer to the other, or from host-to-host, not host-to-client.
-p -parallel parallelism	Specifies number of parallel data connections to use. GridFTP defines multiple wire modes for the data channel. The stream mode flows bytes in order over a single TCP connection; MODE E, or the Extended Block Mode, sends data in blocks, with each block consisting of eight bits of flags, a 64 bit integer indicating the offset from the start of the transfer, a 64 bit integer indicating the length of the block in bytes, and a payload of length bytes. Because offset and length are provided, out of order arrival is acceptable, allowing use of multiple TCP channels. If you use this option, globus-url-copy automatically puts servers into MODE E . Note that using -p 1 is not equivalent to using no parallel option. In both cases, transfers use a single stream; however, the no parallel option uses stream mode, while -p 1 uses MODE E NOTE When choosing a value, start by specifying four streams; four to eight streams is usually sufficient. Several factors affect the actual ideal number, including bandwidth, RTT, and network congestion. If eight streams do not handle the transfer gracefully, increase the value to 16; values over 16 are not efficient
-bs -block-size block	Specifies size, in bytes, of buffer to be used.
-s -subject subject	Specifies a subject to match with both source and destination servers.
-ss -source-subject	Specifies a subject to match with the source server.

Table 7. Options for globus-urlopy

Option	Description
-ds -dest-subject	Specifies a subject to match with the destination server.
-nodcau -no-data-channel-authentication	Turns off data channel authentication; default is data channel authentication. NOTE Turning off data channel authentication is a security risk and is not recommended.
-dcsafe -data-channel-safe	Sets data channel protection mode to SAFE. Guarantees data channel has not been altered, though it can be observed. NOTE Using this option results in an appreciable performance penalty and is not recommended for general use.
-dcpriv -data-channel-private	Sets data channel protection mode to PRIVATE. Guarantees data channel has not been altered; if observed, it will not be understandable. NOTE Using this option results in a substantial performance penalty and is not recommended for general use.

Chapter 6: Using WS-GRAM

This chapter provides information about using WS-GRAM. WS-GRAM is run as part of the web services container, which is started only on the cluster's main node. You may use the main node WS-GRAM service from workstation and login nodes by passing in the **-factory** option, along with the correct service URL, when running **globusrun-ws**; details on using **globusrun-ws** options are in the table "Options for globusrun-ws."

In WS-GRAM, a job is defined as a computational task that may be scheduled on multiple nodes. A single job may require coordinated staging of data and retrieval of results, during, and after execution. WS-GRAM combines job-management services and local system adapters with other service components to support job execution with coordinated file staging.

Throughout this chapter, *install-dir* refers to the UniCluster Express installation location. The environmental variable, **\$GLOBUS_LOCATION**, points to the UniCluster Express installation directory and is used by Globus Toolkit components. Where required, you may set this environmental variable, or enter the complete path to the UniCluster Express installation location.

Using globusrun-ws

globusrun-ws is the official job submission tool for WS-GRAM. Syntax for **globusrun-ws** is:

```
globusrun-ws -option -option ...
```

For example:

```
[user@host ~]$ source &lt;install-dir&gt;/unicluster-user-env.sh
[user@host ~]$ unicluster-grid-login
[user@host ~]$ globusrun-ws -submit -factory
https://mainnode.univaud.com:8443/
wsrf/services/ManagedJobFactoryService -c /bin/true
Submitting job...Done.
Job ID: uuid:992864c6-711e-11dc-9191-00145e0b9108
Termination time: 10/03/2007 19:35 GMT
Current job state: Active
Current job state: CleanUp
```

Current job state: Done
 Destroying job...Done.

This command submits a simple job remotely from the host to the main node. Valid options for **globusrun-ws** are in the table “Options for globusrun-ws.”

Table 8.

Option	Description
<i>-help</i>	Displays help.
<i>-usage</i>	Outputs brief command usage information.
<i>-host</i> <i>-host-authz</i>	GSI host authorization is used to verify the service is using a host credential appropriate for the underlying service address information; this is the default.
<i>-self</i> <i>-self-authz</i>	GSI self authorization is used to verify the service is using a proxy credential derived from the same identity as the client's.
<i>-subject</i> <i>-subject-authz</i> > <i>subjectName</i>	If supplied, the service must be using a credential with the exact subject name provided in <i>subjectName</i> .
<i>-b</i> <i>-batch</i>	Enables batch mode; default is interactive in which the tool keeps running to monitor job status. In batch mode, the tool prints the EPR and exits.
<i>-s</i> <i>-streaming</i>	Sends output to stdout or stderr. Standard output contains only job output data, while standard error may be a mix of errors and messages. Streaming depends on the ability to access job outputs via GridFTP. If -streaming is selected and the job description does not specify an output file redirection, globusrun-ws adds unique output file name redirections and automatic cleanup directives to the job description. If you are using -batch, include -streaming, since it always introduces a 'CleanUp Hold' state which ensures all data is streamed before files are destroyed. Using -streaming with -batch requires use of -monitor so the hold is released. This option implies -staging-delegate if stdout and stderr are not specified.
<i>-so</i> <i>-stdout-file filename</i>	Appends stdout out stream to specified file instead of to stdout.
<i>-se</i> <i>-stderr-file filename</i>	Appends stderr out stream to specified file instead of to stderr.
<i>-ipv6</i> <i>-allow-ipv6</i>	Allows streaming transfers to use IPV6.
<i>-passive</i>	Forces streaming transfers to use MODE S to allow passive mode transfers. This option may be useful within a firewalled environment, however, it does not allow connection caching.

Table 8.

Option	Description
<i>-nodcau</i>	Disables data channel authentication on streaming transfers.
<i>-Ff -factory-epr-file filename</i>	Causes the EPR for the ManagedJobFactory to be read from the given file; this EPR is used as the service endpoint for job submission. Addressing information for the target of this submission must be provided.
<i>-F -factory contact</i>	Causes an EPR to be constructed using ad-hoc methods that depend on UniCluster Express implementation details. For interoperability, it is recommended that -factory-epr-file be used instead. Default values form the following contact information if not overridden: <code>https://localhost:8443/wsrf/services/ManagedJobFactoryService</code>
<i>-Ft -factory-type type</i>	In the absence of -factory-epr-file , this option refines the behavior of the -factory to select a specific type of scheduler; the default is fork for single jobs and multi for multijobs. Use this option with SGE to submit jobs to Grid Engine.
<i>-f -job-description-file filename</i>	Causes the job description to be read from the given file; this description is modified according to other options and is passed WS-GRAM submission messages. The root element of this file must be job for a single job or multiJob for a multijob.
<i>-c -job-command [--] prog [arg...]</i>	Takes all remaining globusrun-ws arguments as its arguments; therefore, it must appear last. This option causes globusrun-ws to generate a simple job description with the named program and arguments.
<i>-I -submission-id ID</i>	Causes job to be submitted using the given ID. NOTE It is an error to use both -I and -If to provide an input ID.
<i>-If -submission-id-file filename</i>	Causes the ID to be read from filename. NOTE It is an error to use both -I and -If to provide an input ID.
<i>-Io -submission-id-output-file filename</i>	Causes the ID in use to be written to the given file, whether the ID was provided by the user or given by one of the above input options.
<i>-J -job-delegate</i>	If the job description does not already provide a jobCredential element, this option forces globusrun-ws to delegate the client credential to WS-GRAM and introduce the corresponding element to the submission input.

Table 8.

Option	Description
<i>-S -staging-delegate</i>	If the job description includes staging or cleanup directives and the job description does not already provide the necessary stagingCredential or transferCredential elements, this option forces globusrun-ws to delegate client credentials to WS-GRAM and RFT and introduce corresponding elements to the submission input. This option is implied by -streaming .
<i>-Jf -job-credential-file</i>	If the job description does not already provide a jobCredential element, this option forces globusrun-ws to copy the supplied EPR into the job description. This should be an EPR returned from the DelegationFactoryService intended for use by the job (or, in the case of multijobs, for authenticating to subjobs). NOTE For multijob descriptions, only the top level jobCredential is copied.
<i>-Sf -staging-credential-file filename</i>	If the job description does not already provide a stagingCredential element, this option forces globusrun-ws to copy the supplied EPR into the job description. This should be DelegationFactoryService intended for use with the RFT service associated with the ManagedJobService. NOTE This option ignored for multijobs
<i>-Tf -transfer-credential-file</i>	Causes globusrun-ws to copy the EPR into each of the stage in, stage out, and cleanup elements that do not already contain a transferCredential element. This should be an EPR returned from the DelegationFactoryService intended for use by RFT to authenticate with the target GridFTP server. This option is ignored for multijobs.
<i>-term -termination mm/dd/yyyy HH:MM</i>	Set an absolute termination time.
<i>-term -termination +HH:MM</i>	Set a termination time relative to the successful creation of the job; default is +24:00
<i>-q -quiet</i>	Suppresses all but fatal standard error messages.
<i>-dbg -debug</i>	Displays all SOAP and FTP control messages on stderr.
<i>-p -private</i>	Enables privacy protection between globusrun-ws and WS-GRAM or GridFTP services. NOTE A fatal error occurs if you select privacy protection and it is not available due to build options or other security settings. This option is only supported with HTTPS endpoints.

Table 8.

Option	Description
<i>-T -http-timeout milliseconds</i>	Sets a timeout for the HTTP socket, in milliseconds, for all web services interactions; default is 120000 (2 minutes).
<i>-n -no-cleanup</i>	Disables default behavior of trapping interrupts (SIG_INTR) and cancelling the job; instead, an interrupt simply causes the tool to exit without affecting the resource.
<i>-submit</i>	Submits (or resubmits) a job to a job host using an XML-based job description document. Jobs can be submitted in one of three output modes: batch, <i>interactive</i> or <i>interactive-streaming</i> . A description of the job must be provided, either using the WS-GRA XML description syntax, or UNIX command and argument list.
<i>-o -job-epr-output-file filename</i>	Writes the ManagedJob EPR to filename following successful submission.
<i>-validate</i>	Checks the job description for syntax errors and a subset of semantic errors without making any service requests.
<i>-f -job-description-file filename</i>	Causes the job description to be read from filename; this description is checked for validity. Addressing information for the ManagedJob target of this option must be provided.
<i>-monitor</i>	Attaches to an existing job in <i>interactive</i> or <i>interactive-streaming</i> output modes.
<i>-j -job-epr-file filename</i>	Causes the EPR for the ManagedJob to be read from the given file. This EPR is used as the endpoint for service requests.
<i>-s -streaming</i>	Enables streaming mode
<i>-status</i>	Reports current state of the job and exits.
<i>-kill</i>	Requests immediate cancellation of the job and exits.

To use **globusrun-ws** to submit a job to Grid Engine, use the **-factory-type** option:

```
globusrun-ws -submit -factory-type SGE -factory
https://mainnode.domain.com -c /bin/false
Submitting job...Done.
Job ID: uuid:bd7cdf02-07a5-11db-a872-000c291538b2
Termination time: 10/30/2007 19:30 GMT
Current job state: Pending
Current job state: Active
Current job state: CleanUp
Current job state: Done
Destroying job...Done.
```


Chapter 7: Using GSI-OpenSSH

This chapter contains information for using GSI-OpenSSH command line tools. GSI-OpenSSH is a modified version of OpenSSH; for additional information on OpenSSH, see the [OpenSSH website](http://www.openssh.org/) [<http://www.openssh.org/>].

GSI-OpenSSH is an SSH server that integrates with the UniCluster Express authorization framework. When GSI-OpenSSH is implemented in the UniCluster Express environment, users can retrieve credentials with **unicluster-grid-logon** and use those credentials to connect to other nodes in the cluster running GSI-OpenSSH without requiring a password.

The default GSI-OpenSSH port used with UniCluster Express is 2222; using this port avoids possible conflicts with existing ssh implementations.

Using GSI-OpenSSH Command Line Tools

GSI-OpenSSH includes three command line tools:

- `gsissh` is a secure remote login;
- `gscp` allows you to securely use remote copy;
- `gsiftp` is a secure file transfer utility.

Using `gsissh`

Syntax of **gsissh** is:

```
gsissh [ -l login_name host.domain.com | user@host.domain.com ]  
-p 2222 command
```

where

- `-l login_name` passes in the username for log in;
- `host.domain.com` is the hostname of the node to which you are connecting;
- `user@host.domain.com` combines the username and hostname for login, and may be used instead of `-l login_name host.domain.com`, as in most standard ssh connections;

- **-p 2222** passes in the default GSI-OpenSSH port (2222) for connection;
- **command** is an optional command that can be executed on the remote machine. When the user's identity has been accepted, the remote machine executes the given command on the remote host instead of a login shell.

For example:

```
[user@hostA ~]$ gsissh -l user hostB.univaud.com -p 2222
Last login: Wed Oct 24 12:43:55 2007 from hostA.univaud.com
Message as of: 01/22/2008
```

```
=====
```

```
This is a UniCluster Express workstation node installation.
```

```
-----
[user@hostB ~]$
```

The session terminates when the command or shell on the remote machine exits, and all X11 and TCP connections have been closed.

Using gsiscp

Syntax of **gsiscp** is:

```
gsiscp -P 2222 /path/to/localFile
user@hostB.domain.com:/path/to/destfile
```

where:

- **-P 2222** passes in the default GSI-OpenSSH port (2222)
- **/path/to/localFile** is the path to the local file that is to be copied;
- **user@hostB.domain.com** is the username and hostname of the destination node;
- **/path/to/destfile** is the copy destination file name.

For example:

```
[user@hostA ~]$ gsiscp -P 2222 ~/foo
user@hostB.univaud.com:/home/user/foo
foo          100%      865      0.8KB/s      00:00
```

Using gsisftp

Syntax of **gsisftp** is:

```
gsisftp -oPort 2222 user@host.domain.com
```

where

- **-oPort 2222** specifies port 2222 is used for the connection;
- **user@host** is the username and hostname to the remote node.

For example:

```
[user@hostA bin]$ gsisftp -oPort=2222 user@hostB.univaud.com  
Connecting to hostB.univaud.com...  
sftp>
```

Once you are connected to the GSI-OpenSSH server, you can use standard interactive commands. As with standard ftp, commands are case insensitive, pathnames containing spaces must be enclosed in quotes, and any special characters in pathnames must be escaped with backslashes (\). Some common interactive commands are listed in the table below:

Interactive Commands used with gsisftp

Table 9.

Command	Description
<i>help ?</i>	Displays help text.
<i>version</i>	Displays protocol version.
<i>Displays protocol version.</i>	Quits the session.
<i>pwd</i>	Displays remote working directory.
<i>cd path</i>	Changes remote directory to path.
<i>lcd path</i>	Changes local directory to path.
<i>lpwd</i>	Prints local working directory.
<i>mkdir path</i>	Creates remote directory specified by path.
<i>lmkdir path</i>	Creates local directory specified by path.

Table 9.

Command	Description
<i>rename oldpath newpath</i>	Renames remote file from oldpath to newpath.
<i>rm path</i>	Deletes remote file specified by path.
<i>rmdir path</i>	Removes remote directory specified by path.
<i>ln oldpath newpath</i>	Creates symbolic link from oldpath to newpath.
<i>symlink oldpath newpath</i>	Creates a symbolic link from oldpath to newpath.
<i>! command</i>	Executes command in local shell.
<i>!</i>	Escapes to local shell.
<i>lls [ls-options [path]]</i>	Displays local directory listing of either path, or current directory if path is not specified; ls-options may contain any flags supported by the local system's ls command.
<i>ls [-laflnrSt] [path]</i>	Displays a remote directory listing of either path or the current directory if path is not specified. The following flags are recognized and alter the behaviour of ls accordingly: -l produces single columnar output. -a lists files beginning with a dot (.). -f specifies the tool does not sort the listing; the default sort order is lexicographical. -l displays additional details including permissions and ownership information. -n produces a long listing with user and group information presented numerically. -r reverses the sort order of the listing. -S sorts the listing by file size. -t sorts the listing by last modification time.
<i>get [-P] remote- path [local-path]</i>	Retrieves the remote-path and store it on the local machine. If the local path name is not specified, it is given the same name it has on the remote machine. If -P is specified, full file permissions and access times are copied as well.
<i>put [-P] local-path [remote-path]</i>	Uploads local-path and store it on the remote machine. If the remote path name is not specified, it is given the same name it has on the local machine. If -P is specified, full file permissions and access times are copied as well.

Chapter 8: Using Reliable File Transfer

In the UniCluster Express environment, Reliable File Transfer (RFT), is used only for file staging on WS-GRAM jobs. There are, however, two command line tools that may be invoked directly to manage data: `rft` and `rft-delete`. RFT functionality is available only on the main node, though you may submit RFT requests remotely from login or workstation nodes.

Throughout this chapter, *install-dir* refers to the UniCluster Express installation location. The environmental variable, **\$GLOBUS_LOCATION**, points to the UniCluster Express installation directory and is used by Globus Toolkit components. Where required, you may set this environmental variable, or enter the complete path to the UniCluster Express installation location.

When invoking the RFT command line tools, you create an RFT resource by submitting a transfer request with an input file; the input file is populated with a set of third party GridFTP URLs for source and destination files.

An example RFT input file is:

```
#true=binary false=ascii
true
#Block size in bytes
16000
#TCP Buffer size in bytes
16000
#Notpt (No thirdPartyTransfer)
false
#Number of parallel streams
1
#Data Channel Authentication (DCAU)
true
# Concurrency of the request
1
#Subject name of source server
/O=UnivaUD/OU=univaud.com/CN=mainnode.univaud.com
#Subject name of destination server
/O=UnivaUD/OU=univaud.com/CN=mainnode.univaud.com
#Transfer all or none of the transfers
false
#Maximum number of retries
10
#Source/Dest URL Pairs
gsiftp://mainnode.domain.com:2811/tmp/newTest
gsiftp://mainnode.domain.com:2811/tmp/testDone
```

For the input file to work correctly, some information in the example file is replaced with system-specific values; these are listed in table

Table 10. Environment-Specific Parameters for the RFT Input File

Parameter	Value
Subject name of source server	In this example, the value is /O=UnivaUD/OU=univaud.com/CN=mainnode.univaud.com; this must be replaced with the main node's service certificate DN. This can be obtained by running: [user@host ~]\$ grid-cert-info -subject -f installldir/etc/gt_ws_cert.pem /O=UnivaUD/OU=univaud.com/CN=mainnode.univaud.com where install-dir refers to the path to the UniCluster Express installation directory.
Subject name of destination server	Since RFT runs within a service container, and the main node is the only node on the cluster to run a container, this subject name should also be that of the main node.
Source URL	In this example, the value is gsiftp://mainnode.domain.com:2811/tmp/newTest; this must be replaced with the complete URL of the source file in the same format: gsiftp://mainnode.domain.com:port//path/to/file.
Destination URL	In this example, the value is gsiftp://mainnode.domain.com:2811/tmp/testDone; this must be replaced with the complete URL of the destination file in the same format gsiftp://mainnode.domain.com:port//path/to/file.

After job submission, client interactions with RFT are much like any other job scheduler and standard WSRF tools can be used to query job status and subscribe to state change notifications; information on WSRF tools is in **APPENDIX A**.

Using rft

rft is a simple command that submits and monitors a third party GridFTP transfer.

Syntax:

```
rft -option -option ... -f filepath
```

where *filepath* is the path to the input file.

For example:

```
[user@host ~]$ rft -h mainnode.univaud.com -f /tmp/m_trans.xfr
Number of transfers in this request: 1
Subscribed for overall status
Termination time to set: 60 minutes
```

```

Overall status of transfer:
Finished/Active/Failed/Retrying/Pending
0/1/0/0/0
Overall status of transfer:
Finished/Active/Failed/Retrying/Pending
1/0/0/0/0
All Transfers are completed.

```

Valid options for <command>rft</command> are in the table below:

Table 11. Options for rft

Option	Description
-h	Host name or IP of the container; default is localhost.
-r	Port on which container is listening; default is 8443.
-l	Lifetime, in minutes, of the created resource; default is 60. NOTE For extremely long transfers, use -l to supply a lifetime long enough to satisfy the transfer time required. For example, for a transfer that will take 10 hours to complete, use rft -l 600 .
-m	Sets security mechanism; allowed values are msg for secure message, conv for secure conversation or trans for transport. Default is trans .
-p	Sets protection type; allowed values are sig for signature or enc for encryption. Default is sig .
-z	Sets authorization mechanism; allowed values are self for self-authorization or host for host authorization. Default is host .
-file	File in which to write the Endpoint Reference (EPR) of the RFT resource created. Using -file makes it possible to cancel an RFT request after the request is made. For example, use rft -file rft_transfer.EPR to write the job request EPR to a file named rft_transfer.EPR . To kill this transfer, use wsrf-destroy -eprFile rft_transfer.EPR
-f	Path to the input file.

Using rft-delete

rft-delete deletes files using RFT entry. The syntax is:

```
rft-delete -option -option ... -f filepath
```

where filepath is the path to the input file.

For example:

```
[user@host ~]$ cat delfile.xfr
# Subject name (defaults to host subject)
/O=UnivaUD/OU=univaud.com/CN=host.univaud.com
gsiftp://localhost:2811/tmp/rftTest01
gsiftp://localhost:2811/tmp/rftTest02
[user@host ~]$ rft-delete -h host.univaud.com -f delfile.xfr
creating a rft resource
Subscribed for overall status
Termination time to set: 60 minutes
Overall status of transfer:
Overall status of transfer:
Finished/Active/Failed/Retrying/Pending
Finished/Active/Failed/Retrying/Pending
0/1/1/0/0/0/0/1
0/1
Overall status of transfer:
Finished/Active/Failed/Retrying/Pending
2/0/0/0/0
Overall status of transfer:
Finished/Active/Failed/Retrying/Pending
1/1/0/0/0
All Transfers are completed
[user@host ~]$ ll /tmp | grep rftTest
```

When using rft-delete, it is important that there are at least two files listed for deletion, and that there are no missed lines. If these criteria are not met, the command displays an error: "Error:[Caused by: Missing '[protocol]://'", even though it may function correctly.

Valid options for rft-delete are in the table given below. :

Table 12. Options for rft-delete

Option	Description
-h	Host name or IP of the container; default is localhost.
-r	Port on which container is listening; default is 8443.
-l	Lifetime, in minutes, of the created resource; default is 60.
	NOTE For extremely long transfers, use -l to supply a lifetime long enough to satisfy the transfer time required. For example, for a transfer that will take 10 hours to complete, use rft -l 600 .

Table 12. Options for rft-delete

Option	Description
-m	Sets security mechanism; allowed values are: msg for secure message; conv for secure conversation; trans for transport. Default is trans.
-p	Sets protection type; allowed values are: sig for signature; enc for encryption. Default is sig.
-z	Sets authorization mechanism; allowed values are: self for self-authorization; host for host authorization. Default is host.
-file	File in which to write the Endpoint Reference (EPR) of the RFT resource created.
	Using -file makes it possible to cancel an RFT request after the request is made. For example, use <code>rft -file rft_transfer.EPR</code> to write the job request EPR to a file named <code>rft_transfer.EPR</code> . To kill this transfer, use <code>wsrf-destroy -eprFile rft_transfer.EPR</code>
-f	Path to the input file.

Chapter 9: Troubleshooting

Throughout this chapter, *install-dir* refers to the UniCluster Express installation location. The environmental variable, *\$GLOBUS_LOCATION* points to the UniCluster Express installation directory and is used by Globus Toolkit components. Where required, you may set this environmental variable, or enter the complete path to the UniCluster Express installation location.

- For a complete list of UniCluster Express FAQs, please refer to the latest information at the Grid.org FAQs page, <http://www.grid.org/faq>.
- Additional information on troubleshooting issues are available on the Grid.org, <http://www.grid.org/wiki/>.
- Access to the latest bugs and fixes is at the Grid.org Bugzilla, <https://bugs.grid.org/>.

Problem:

When I attempt to use **unicluster-grid-logon**, I get an error that says the tool "Failed to obtain GSI credentials."

Solution:

You must source the **unicluster-user-env.sh** file to appropriately configure the environment before you run **unicluster-grid-logon**, or any other UniCluster Express service; refer to [Logging on to the Cluster](#) for details.

Problem:

I can't run `unicluster-user-env.sh`

Solution:

This file is not executable. It is 'sourced' to set up the correct user environment for UniCluster Express functionality. To source this file, bash and sh shell users execute:
[user@host ~]\$. <install-dir>/unicluster-user-env.sh

where *install-dir* is the path to the UniCluster Express installation directory.

Or, csh and tcsh shell users execute:

[user@host ~]\$ source <install-dir>/unicluster-user-env.csh

If you are unsure of which shell you are running, contact your system administrator.

Problem:

When I attempt to log in using `unicluster-grid-logon`, I get this error:

```
[userA@host ~]$ source <install-dir>/unicluster-user-env.sh
[userA@host ~]$ unicluster-grid-logon
Failed to receive credentials.
ERROR from myproxy-server (mainnode.univaud.com):
Mapping call-out returned error
unknown username: userA
Error: unicluster-authn-agent failed
```

where *install-dir* is the path to the UniCluster Express installation directory.

Solution:

This error indicates that you do not have sufficient user accounts and permissions to log in to the cluster. Contact your systems administrator for details.

Appendix A: WSRF and WNS Command Line Tools

This appendix contains information about command line tools for WSRF and WSN components.

WSRF Tools

This section contains standard WSRF command line tools clients can use to manage jobs and query web services.

Using **wsrf-destroy**

wsrf-destroy destroys a resource.

Syntax:

wsrf-destroy -option -option ...

Table 13. Options for “wsrf-destroy”

Option	Description
<i>h</i> <i>--help</i>	Displays help.
<i>-d</i> <i>--debug</i>	Enables debug mode; full stack traces of errors are displayed.
<i>-e</i> <i>--eprFile file</i>	Specifies an XML file that contains the WS-Addressing EPR.
<i>-s</i> <i>--service url</i>	Specifies the service URL
<i>-k</i> <i>--key name</i>	Specifies resource key; <i>name</i> is the qualified name of the resource key in string form, for example: {namespaceURI}localPart. The value of the key is simple; for complex keys, use <i>--eprFile</i> .
<i>-f</i> <i>--descriptor file</i>	Specifies a client security descriptor; overrides all other security settings.

Table 13. Options for “wsrf-destroy”

Option	Description
-a --anonymous	Enables anonymous authentication; only supported with transport security or GSI Secure Conversation authentication mechanism.
-g --delegation mode	Enables delegation; <i>mode</i> is either <i>limited</i> or <i>full</i> . This option only supported with GSI Secure Conversation authentication mechanism.
l --contextLifetime value	Sets lifetime of client security context in milliseconds; this option is only supported with GSI Secure Conversation authentication mechanism.
-m --securityMech type	Specifies authentication mechanism: <i>msg</i> for GSI Secure Message or <i>conv</i> for GSI Secure Conversation.
-c --serverCertificate file	Specifies server's certificate file used for encryption; only needed for GSI Secure Message authentication mechanism.
-p --protection type	Specifies protection level: <i>sig</i> for signature or <i>encfor</i> encryption.
-x --proxyFilename value	Sets the proxy certificate file to use as client credential.
-z --authorization type	Specifies authorization type; <i>type</i> is <i>self</i> , <i>host</i> , or <i>none</i> or a string specifying the expected identity of the remote party.
-t --timeout time	Specifies client timeout in seconds; default is <i>10</i> . The client waits for <i>time</i> seconds for a response from the server before returning an error.

Example of Use

```
wsrf-destroy -s
http://localhost:8443/wsrf/services/CounterService
"{http://counter.com};CounterKey" 123
```

Using wsrf-set-termination-time

wsrf-set-termination-time

Syntax:

```
wsrf-set-termination-time -option -option ... time
```

where *time* is either the number of seconds or *infinity*.

Table 14. Options for “wsrf-set-termination-time”

Option	Description
<i>h</i> --help	Displays help.
<i>-d</i> --debug	Enables debug mode; full stack traces of errors are displayed.
<i>-e</i> --eprFile file	Specifies an XML file that contains the WS-Addressing EPR.
<i>-s</i> --service url	Specifies the service URL
<i>-k</i> --key name	Specifies resource key; <i>name</i> is the qualified name of the resource key in string form, for example: {namespaceURI}localPart. The value of the key is simple; for complex keys, use <i>--eprFile</i> .
<i>-f</i> --descriptor file	Specifies a client security descriptor; overrides all other security settings.
<i>-a</i> --anonymous	Enables anonymous authentication; only supported with transport security or GSI Secure Conversation authentication mechanism.
<i>-g</i> --delegation mode	Enables delegation; <i>mode</i> is either <i>limited</i> or <i>full</i> . This option only supported with GSI Secure Conversation authentication mechanism.
<i>l</i> --contextLifetime value	Sets lifetime of client security context in milliseconds; this option is only supported with GSI Secure Conversation authentication mechanism.
<i>-m</i> --securityMech type	Specifies authentication mechanism: <i>msg</i> for GSI Secure Message or <i>conv</i> for GSI Secure Conversation.
<i>-c</i> --serverCertificate file	Specifies server's certificate file used for encryption; only needed for GSI Secure Message authentication mechanism.
<i>-p</i> --protection type	Specifies protection level: <i>sig</i> for signature or <i>encfor</i> for encryption.
<i>-x</i> --proxyFilename value	Sets the proxy certificate file to use as client credential.
<i>-z</i> --authorization type	Specifies authorization type; <i>type</i> is <i>self</i> , <i>host</i> , or <i>none</i> or a string specifying the expected identity of the remote party.
<i>-t</i> --timeout time	Specifies client timeout in seconds; default is <i>10</i> . The client waits for <i>time</i> seconds for a response from the server before returning an error.

Example of Use

```
wsrf-set-termination-time -s
http://localhost:8443/wsrf/services/CounterService
```

```
\ -k "{http://counter.com}CounterKey" 123 30
```

Using wsrf-query

wsrf-query performs query on a resource property document.

Syntax:

```
wsrf-query -option -option ... queryExp dialect
```

where:

- *queryExp* is a query expression;
- *dialect* is the query dialect.

Table 15. Options for “wsrf-query-options”

Option	Description
<i>h</i> --help	Displays help.
<i>-d</i> --debug	Enables debug mode; full stack traces of errors are displayed.
<i>-e</i> --eprFile file	Specifies an XML file that contains the WS-Addressing EPR.
<i>-s</i> --service url	Specifies the service URL
<i>-k</i> --key name	Specifies resource key; <i>name</i> is the qualified name of the resource key in string form, for example: {namespaceURI}localPart. The value of the key is simple; for complex keys, use <i>--eprFile</i> .
<i>-f</i> --descriptor file	Specifies a client security descriptor; overrides all other security settings.
<i>-a</i> --anonymous	Enables anonymous authentication; only supported with transport security or GSI Secure Conversation authentication mechanism.
<i>-g</i> --delegation mode	Enables delegation; <i>mode</i> is either <i>limited</i> or <i>full</i> . This option only supported with GSI Secure Conversation authentication mechanism.
<i>l</i> --contextLifetime value	Sets lifetime of client security context in milliseconds; this option is only supported with GSI Secure Conversation authentication mechanism.
<i>-m</i> --securityMech type	Specifies authentication mechanism: <i>msg</i> for GSI Secure Message or <i>conv</i> for GSI Secure Conversation.
<i>-c</i> --serverCertificate file	Specifies server's certificate file used for encryption; only needed for GSI Secure Message authentication mechanism.

Table 15. Options for “wsrf-query-options”

Option	Description
-p --protection <i>type</i>	Specifies protection level: sig for signature or encfor encryption.
-x --proxyFilename <i>value</i>	Sets the proxy certificate file to use as client credential.
-z --authorization <i>type</i>	Specifies authorization type; <i>type</i> is <i>self</i> , <i>host</i> , or <i>none</i> or a string specifying the expected identity of the remote party.
-t --timeout <i>time</i>	Specifies client timeout in seconds; default is 10. The client waits for <i>time</i> seconds for a response from the server before returning an error.

Examples of Use

```
wsrf-query -s
https://123.456.78.90:8443/wsrf/services/DefaultIndexService \
"count(//*[local-name()='Entry'])"
```

```
wsrf-query -s http://mainnode.domain.com:8443/wsrf/services/
ContainerRegistryService \ "/*/*/*/*[local-name()='Address']"
```

In these examples, *123.456.78.90* must be replaced by the main node IP address; likewise, *mainnode.domain.com* must be replaced by the main node's FQDN.

Using wsrf-get-property

wsrf-get-property gets the values of a single resource property.

Syntax:

```
wsrf-get-property -option -option ... property
```

where *property* is a qualified name of the resource property in string form: *{namespaceURI}localPart..*

Table 16. Options for “wsrf-get-property”

Option	Description
-h --help	Displays help.
-d --debug	Enables debug mode; full stack traces of errors are displayed.
-e --eprFile <i>file</i>	Specifies an XML file that contains the WS-Addressing EPR.
-s --service <i>url</i>	Specifies the service URL

Table 16. Options for “wsrf-get-property”

Option	Description
-k --key <i>name</i>	Specifies resource key; <i>name</i> is the qualified name of the resource key in string form, for example: {namespaceURI}localPart. The value of the key is simple; for complex keys, use <i>--eprFile</i> .
-f --descriptor <i>file</i>	Specifies a client security descriptor; overrides all other security settings.
-a --anonymous	Enables anonymous authentication; only supported with transport security or GSI Secure Conversation authentication mechanism.
-g --delegation <i>mode</i>	Enables delegation; <i>mode</i> is either <i>limited</i> or <i>full</i> . This option only supported with GSI Secure Conversation authentication mechanism.
l --contextLifetime <i>value</i>	Sets lifetime of client security context in milliseconds; this option is only supported with GSI Secure Conversation authentication mechanism.
-m --securityMech <i>type</i>	Specifies authentication mechanism: <i>msg</i> for GSI Secure Message or <i>conv</i> for GSI Secure Conversation.
-c --serverCertificate <i>file</i>	Specifies server's certificate file used for encryption; only needed for GSI Secure Message authentication mechanism.
-p --protection <i>type</i>	Specifies protection level: <i>sig</i> for signature or <i>encfor</i> for encryption.
-x --proxyFilename <i>value</i>	Sets the proxy certificate file to use as client credential.
-z --authorization <i>type</i>	Specifies authorization type; <i>type</i> is <i>self</i> , <i>host</i> , or <i>none</i> or a string specifying the expected identity of the remote party.
-t --timeout <i>time</i>	Specifies client timeout in seconds; default is <i>10</i> . The client waits for <i>time</i> seconds for a response from the server before returning an error.

Example of Use

```
wsrf-get-property -s
http://localhost:8443/wsrf/services/CounterService
\ -k "{;http://counter.com};CounterKey" 123 \
"{;http://docs.oasis-open.org/wsrf/
2004/06/wsrf-WS-ResourceLifetime-1.2-draft-01.xsd};CurrentTime"
```

Using wsrf-get-properties

wsrf-get-properties gets values of multiple resource properties.

Syntax:

```
wsrf-get-properties -option -option ... property property ...
```

where *property* is a qualified name of the resource property in string form:
`{namespaceURI}localPart`.

Table 17. Options for “wsrf-get-properties”

Option	Description
<i>h</i> --help	Displays help.
<i>-d</i> --debug	Enables debug mode; full stack traces of errors are displayed.
<i>-e</i> --eprFile file	Specifies an XML file that contains the WS-Addressing EPR.
<i>-s</i> --service url	Specifies the service URL
<i>-k</i> --key name	Specifies resource key; <i>name</i> is the qualified name of the resource key in string form, for example: <code>{namespaceURI}localPart</code> . The value of the key is simple; for complex keys, use <code>--eprFile</code> .
<i>-f</i> --descriptor file	Specifies a client security descriptor; overrides all other security settings.
<i>-a</i> --anonymous	Enables anonymous authentication; only supported with transport security or GSI Secure Conversation authentication mechanism.
<i>-g</i> --delegation mode	Enables delegation; <i>mode</i> is either <i>limited</i> or <i>full</i> . This option only supported with GSI Secure Conversation authentication mechanism.
<i>l</i> --contextLifetime value	Sets lifetime of client security context in milliseconds; this option is only supported with GSI Secure Conversation authentication mechanism.
<i>-m</i> --securityMech type	Specifies authentication mechanism: <i>msg</i> for GSI Secure Message or <i>conv</i> for GSI Secure Conversation.
<i>-c</i> --serverCertificate file	Specifies server's certificate file used for encryption; only needed for GSI Secure Message authentication mechanism.
<i>-p</i> --protection type	Specifies protection level: <i>sig</i> for signature or <i>encfor</i> encryption.
<i>-x</i> --proxyFilename value	Sets the proxy certificate file to use as client credential.
<i>-z</i> --authorization type	Specifies authorization type; <i>type</i> is <i>self</i> , <i>host</i> , or <i>none</i> or a string specifying the expected identity of the remote party.
<i>-t</i> --timeout time	Specifies client timeout in seconds; default is <i>10</i> . The client waits for <i>time</i> seconds for a response from the server before returning an error.

Example of Use

```
wsrf-get-properties -s
://localhost:8443/wsrf/services/CounterService
\ -k "{;http://counter.com};CounterKey" 123 \
"{;http://docs.oasis-open.org/
wsrf/2004/06/wsrf-WS-ResourceLifetime-1.2-draft
01.xsd};CurrentTime"
\ "{;http://docs.oasis-open.org/wsrf/2004/06
wsrf-WS-ResourceLifetime-1.2-draft
01.xsd};TerminationTime"
```

Using wsrf-insert-property

wsrf-insert-property inserts values into a resource property.

Syntax:

```
wsrf-insert-property -option -option ... propValueFile
```

where *propValueFile* is an XML file containing the value of the resource property with the qualified name of the property being the outer most element.

Table 18. Options for “wsrf-insert-property”

Option	Description
<i>h</i> <i>--help</i>	Displays help.
<i>-d</i> <i>--debug</i>	Enables debug mode; full stack traces of errors are displayed.
<i>-e</i> <i>--eprFile file</i>	Specifies an XML file that contains the WS-Addressing EPR.
<i>-s</i> <i>--service url</i>	Specifies the service URL
<i>-k</i> <i>--key name</i>	Specifies resource key; <i>name</i> is the qualified name of the resource key in string form, for example: {namespaceURI}localPart. The value of the key is simple; for complex keys, use <i>--eprFile</i> .
<i>-f</i> <i>--descriptor file</i>	Specifies a client security descriptor; overrides all other security settings.
<i>-a</i> <i>--anonymous</i>	Enables anonymous authentication; only supported with transport security or GSI Secure Conversation authentication mechanism.
<i>-g</i> <i>--delegation mode</i>	Enables delegation; <i>mode</i> is either <i>limited</i> or <i>full</i> . This option only supported with GSI Secure Conversation authentication mechanism.

Table 18. Options for “wsrf-insert-property”

Option	Description
<i>l</i> --contextLifetime <i>value</i>	Sets lifetime of client security context in milliseconds; this option is only supported with GSI Secure Conversation authentication mechanism.
-m --securityMech <i>type</i>	Specifies authentication mechanism: <i>msg</i> for GSI Secure Message or <i>conv</i> for GSI Secure Conversation.
-c --serverCertificate <i>file</i>	Specifies server's certificate file used for encryption; only needed for GSI Secure Message authentication mechanism.
-p --protection <i>type</i>	Specifies protection level: <i>sig</i> for signature or <i>encfor</i> for encryption.
-x --proxyFilename <i>value</i>	Sets the proxy certificate file to use as client credential.
-z --authorization <i>type</i>	Specifies authorization type; <i>type</i> is <i>self</i> , <i>host</i> , or <i>none</i> or a string specifying the expected identity of the remote party.
-t --timeout <i>time</i>	Specifies client timeout in seconds; default is 10. The client waits for <i>time</i> seconds for a response from the server before returning an error.

Example of Use

```
wsrf-insert-property -s
http://localhost:8443/wsrf/services/WidgetService
\ -k
"{;http://www.globus.org/namespaces/2004/06/core};WidgetKey};"
123 \ in.xml
```

where *in.xml* looks like:

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

Using wsrf-update-property

wsrf-update-property updates value of a resource property.

Syntax:

```
wsrf-update-property -option -option ... propValueFile
```

where *propValueFile* is an XML file containing the value of the resource property with the qualified name of the property being the outer most element.

Table 19. Options for “wsrf-update-property”

Option	Description
<i>h</i> --help	Displays help.
<i>-d</i> --debug	Enables debug mode; full stack traces of errors are displayed.
<i>-e</i> --eprFile file	Specifies an XML file that contains the WS-Addressing EPR.
<i>-s</i> --service url	Specifies the service URL
<i>-k</i> --key name	Specifies resource key; <i>name</i> is the qualified name of the resource key in string form, for example: {namespaceURI}localPart. The value of the key is simple; for complex keys, use --eprFile.
<i>-f</i> --descriptor file	Specifies a client security descriptor; overrides all other security settings.
<i>-a</i> --anonymous	Enables anonymous authentication; only supported with transport security or GSI Secure Conversation authentication mechanism.
<i>-g</i> --delegation mode	Enables delegation; <i>mode</i> is either <i>limited</i> or <i>full</i> . This option only supported with GSI Secure Conversation authentication mechanism.
<i>l</i> --contextLifetime value	Sets lifetime of client security context in milliseconds; this option is only supported with GSI Secure Conversation authentication mechanism.
<i>-m</i> --securityMech type	Specifies authentication mechanism: <i>msg</i> for GSI Secure Message or <i>conv</i> for GSI Secure Conversation.
<i>-c</i> --serverCertificate file	Specifies server's certificate file used for encryption; only needed for GSI Secure Message authentication mechanism.
<i>-p</i> --protection type	Specifies protection level: <i>sig</i> for signature or <i>encfor</i> for encryption.
<i>-x</i> --proxyFilename value	Sets the proxy certificate file to use as client credential.
<i>-z</i> --authorization type	Specifies authorization type; <i>type</i> is <i>self</i> , <i>host</i> , or <i>none</i> or a string specifying the expected identity of the remote party.
<i>-t</i> --timeout time	Specifies client timeout in seconds; default is 10. The client waits for <i>time</i> seconds for a response from the server before returning an error.

Example of Use

```
wsrf-update-property -s
http://localhost:8443/wsrf/services/WidgetService
\ -k
"{;http://www.globus.org/namespaces/2004/06/core};WidgetKey"
123 \ in.xml
```

where *in.xml* looks like:

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

Using wsrf-delete-property

wsrf-delete-property deletes a resource property.

Syntax:

```
wsrf-delete-property -option -option ... property
```

where *property* is a qualified name of the resource property in string form:
{namespaceURI}localPart.

Table 20. Options for “wsrf-delete-property”

Option	Description
<i>h</i> --help	Displays help.
<i>-d</i> --debug	Enables debug mode; full stack traces of errors are displayed.
<i>-e</i> --eprFile file	Specifies an XML file that contains the WS-Addressing EPR.
<i>-s</i> --service url	Specifies the service URL
<i>-k</i> --key name	Specifies resource key; <i>name</i> is the qualified name of the resource key in string form, for example: <i>{namespaceURI}localPart</i> . The value of the key is simple; for complex keys, use <i>--eprFile</i> .
<i>-f</i> --descriptor file	Specifies a client security descriptor; overrides all other security settings.
<i>-a</i> --anonymous	Enables anonymous authentication; only supported with transport security or GSI Secure Conversation authentication mechanism.
<i>-g</i> --delegation mode	Enables delegation; <i>mode</i> is either <i>limited</i> or <i>full</i> . This option only supported with GSI Secure Conversation authentication mechanism.

Table 20. Options for “wsrf-delete-property”

Option	Description
<i>l</i> --contextLifetime <i>value</i>	Sets lifetime of client security context in milliseconds; this option is only supported with GSI Secure Conversation authentication mechanism.
-m --securityMech <i>type</i>	Specifies authentication mechanism: <i>msg</i> for GSI Secure Message or <i>conv</i> for GSI Secure Conversation.
-c --serverCertificate <i>file</i>	Specifies server's certificate file used for encryption; only needed for GSI Secure Message authentication mechanism.
-p --protection <i>type</i>	Specifies protection level: <i>sig</i> for signature or <i>encfor</i> for encryption.
-x --proxyFilename <i>value</i>	Sets the proxy certificate file to use as client credential.
-z --authorization <i>type</i>	Specifies authorization type; <i>type</i> is <i>self</i> , <i>host</i> , or <i>none</i> or a string specifying the expected identity of the remote party.
-t --timeout <i>time</i>	Specifies client timeout in seconds; default is <i>10</i> . The client waits for <i>time</i> seconds for a response from the server before returning an error.

Example of Use

```
wsrf-delete-property -s
http://localhost:8443/wsrf/services/WidgetService
\ -k "{http://www.globus.org/namespaces/2004/06/core}WidgetKey"
123 \ "{http://widgets.com};foo"
```

WS-Notification Tools

This section contains standard WSN command line tools clients can use to manage web services notification subscriptions.

Using wsn-get-current-message

wsn-get-current-message gets a current message associated with a specific topic.

Syntax:

```
wsn-get-current-message -option -option ... topic
```

where *topic* is a qualified name of the resource property in the form *{namespaceURI}localPart*.

Table 21. Options for “wsn-get-current-message”

Option	Description
<i>h</i> --help	Displays help.
<i>-d</i> --debug	Enables debug mode; full stack traces of errors are displayed.
<i>-e</i> --eprFile file	Specifies an XML file that contains the WS-Addressing EPR.
<i>-s</i> --service url	Specifies the service URL
<i>-k</i> --key name	Specifies resource key; <i>name</i> is the qualified name of the resource key in string form, for example: <i>{namespaceURI}localPart</i> . The value of the key is simple; for complex keys, use <i>--eprFile</i> .
<i>-f</i> --descriptor file	Specifies a client security descriptor; overrides all other security settings.
<i>-a</i> --anonymous	Enables anonymous authentication; only supported with transport security or GSI Secure Conversation authentication mechanism.
<i>-g</i> --delegation mode	Enables delegation; <i>mode</i> is either <i>limited</i> or <i>full</i> . This option only supported with GSI Secure Conversation authentication mechanism.
<i>l</i> --contextLifetime value	Sets lifetime of client security context in milliseconds; this option is only supported with GSI Secure Conversation authentication mechanism.
<i>-m</i> --securityMech type	Specifies authentication mechanism: <i>msg</i> for GSI Secure Message or <i>conv</i> for GSI Secure Conversation.
<i>-c</i> --serverCertificate file	Specifies server's certificate file used for encryption; only needed for GSI Secure Message authentication mechanism.
<i>-p</i> --protection type	Specifies protection level: <i>sig</i> for signature or <i>encfor</i> for encryption.
<i>-z</i> --authorization type	Specifies authorization type; <i>type</i> is <i>self</i> , <i>host</i> , or <i>none</i> or a string specifying the expected identity of the remote party.

Example of Use

```
wsn-get-current-message -s
://localhost:8443/wsrf/services/CounterService \
-k "{http://counter.com}CounterKey" 123 \
"{http://counter.com}Value"
```

Using wsn-pause-subscription

wsn-pause-subscription pauses a subscription.

Syntax:

wsn-pause-subscription -option -option ...

Table 22. Options for “wsn-pause-subscription”

Option	Description
<i>h</i> --help	Displays help.
-d --debug	Enables debug mode; full stack traces of errors are displayed.
-e --eprFile <i>file</i>	Specifies an XML file that contains the WS-Addressing EPR.
-s --service <i>url</i>	Specifies the service URL
-k --key <i>name</i>	Specifies resource key; <i>name</i> is the qualified name of the resource key in string form, for example: {namespaceURI}localPart. The value of the key is simple; for complex keys, use --eprFile .
-f --descriptor <i>file</i>	Specifies a client security descriptor; overrides all other security settings.
-a --anonymous	Enables anonymous authentication; only supported with transport security or GSI Secure Conversation authentication mechanism.
-g --delegation <i>mode</i>	Enables delegation; <i>mode</i> is either <i>limited</i> or <i>full</i> . This option only supported with GSI Secure Conversation authentication mechanism.
l --contextLifetime <i>value</i>	Sets lifetime of client security context in milliseconds; this option is only supported with GSI Secure Conversation authentication mechanism.
-m --securityMech <i>type</i>	Specifies authentication mechanism: <i>msg</i> for GSI Secure Message or <i>conv</i> for GSI Secure Conversation.
-c --serverCertificate <i>file</i>	Specifies server's certificate file used for encryption; only needed for GSI Secure Message authentication mechanism.
-p --protection <i>type</i>	Specifies protection level: <i>sig</i> for signature or <i>encfor</i> for encryption.
-z --authorization <i>type</i>	Specifies authorization type; <i>type</i> is <i>self</i> , <i>host</i> , or <i>none</i> or a string specifying the expected identity of the remote party.

Example of Use

```
wsn-pause-subscription -s http://localhost:8443/wsrf/services/
SubscriptionManagerService \ -k "{http://www.globus.org/
namespaces/2004/06/core}acc271c0-4df9-11d9-
ab19-87da3bc7cf28"
```

Using wsn-resume-subscription

wsn-resume-subscription resumes a subscription.

Syntax:

```
wsn-resume-subscription -option -option ...
```

Table 23. Options for “wsn-resume subscription”

Option	Description
<i>h</i> --help	Displays help.
-d --debug	Enables debug mode; full stack traces of errors are displayed.
-e --eprFile <i>file</i>	Specifies an XML file that contains the WS-Addressing EPR.
-s --service <i>url</i>	Specifies the service URL
-k --key <i>name</i>	Specifies resource key; <i>name</i> is the qualified name of the resource key in string form, for example: {namespaceURI}localPart. The value of the key is simple; for complex keys, use --eprFile .
-f --descriptor <i>file</i>	Specifies a client security descriptor; overrides all other security settings.
-a --anonymous	Enables anonymous authentication; only supported with transport security or GSI Secure Conversation authentication mechanism.
-g --delegation <i>mode</i>	Enables delegation; <i>mode</i> is either <i>limited</i> or <i>full</i> . This option only supported with GSI Secure Conversation authentication mechanism.
l --contextLifetime <i>value</i>	Sets lifetime of client security context in milliseconds; this option is only supported with GSI Secure Conversation authentication mechanism.
-m --securityMech <i>type</i>	Specifies authentication mechanism: <i>msg</i> for GSI Secure Message or <i>conv</i> for GSI Secure Conversation.
-c --serverCertificate <i>file</i>	Specifies server's certificate file used for encryption; only needed for GSI Secure Message authentication mechanism.

Table 23. Options for “wsn-resume subscription”

Option	Description
-p --protection <i>type</i>	Specifies protection level: sig for signature or encfor encryption.
-z --authorization <i>type</i>	Specifies authorization type; <i>type</i> is <i>self</i> , <i>host</i> , or <i>none</i> or a string specifying the expected identity of the remote party.

Example of Use

```
wsn-resume-subscription -s
http://localhost:8443/wsrf/services/SubscriptionManagerService
\
-k "{http://www.globus.org/namespaces/2004/06/core}acc271c0-
4df9-
11d9-ab19-87da3bc7cf28"
```

Using wsn-subscribe

wsn-subscribe subscribes to a specific topic.

Syntax:

```
wsn-subscribe -option -option ... topic
```

where *topic* is a qualified name of the resource property in the form *{namespaceURI}localPart*.

Table 24. Options for “wsn-subscribe”

Option	Description
-h --help	Displays help.
-d --debug	Enables debug mode; full stack traces of errors are displayed.
-e --eprFile <i>file</i>	Specifies an XML file that contains the WS-Addressing EPR.
-s --service <i>url</i>	Specifies the service URL
-k --key <i>name</i>	Specifies resource key; <i>name</i> is the qualified name of the resource key in string form, for example: <i>{namespaceURI}localPart</i> . The value of the key is simple; for complex keys, use <i>--eprFile</i> .
-f --descriptor <i>file</i>	Specifies a client security descriptor; overrides all other security settings.

Table 24. Options for “wsn-subscribe”

Option	Description
-a --anonymous	Enables anonymous authentication; only supported with transport security or GSI Secure Conversation authentication mechanism.
-g --delegation mode	Enables delegation; <i>mode</i> is either <i>limited</i> or <i>full</i> . This option only supported with GSI Secure Conversation authentication mechanism.
l --contextLifetime value	Sets lifetime of client security context in milliseconds; this option is only supported with GSI Secure Conversation authentication mechanism.
-m --securityMech type	Specifies authentication mechanism: <i>msg</i> for GSI Secure Message or <i>conv</i> for GSI Secure Conversation.
-c --serverCertificate file	Specifies server's certificate file used for encryption; only needed for GSI Secure Message authentication mechanism.
-p --protection type	Specifies protection level: <i>sig</i> for signature or <i>encfor</i> for encryption.
-z --authorization type	Specifies authorization type; <i>type</i> is <i>self</i> , <i>host</i> , or <i>none</i> or a string specifying the expected identity of the remote party.

Example of Use

```
wsn-subscribe -s
http://localhost:8443/wsrf/services/CounterService
\ -k "{http://counter.com}CounterKey" 123 \
"{http://counter.com}Value"
```


Glossary

X.509 proxy certificate

A widely used standard for defining digital certificates.

access list

A list of users and UNIX groups who are permitted (or denied) access to a resource; also called an ‘access control list’, or ACL.

administration host

A Grid Engine host, other than the master host, that allows you to perform administrative tasks. See also: [execution host](#), [master host](#), [submit host](#).

array job

A Grid Engine job made up of a range of independent identical tasks with each task is a separate job. Array job tasks have unique task identifiers to differentiate them.

asymmetric cryptography

Another name for Public Key Infrastructure. See also: [Public Key Infrastructure \(PKI\)](#).

batch job

A Grid Engine session consisting of a UNIX shell script that can be run without user intervention. See also: [interactive job](#).

cell

In Grid Engine, a cell is a separate cluster with a separate configuration and master; cells can be used to loosely couple separate administrative units.

Certificate Authority (CA)

An entity which issues digital certificates for use by other parties to verify their credentials; also known as Certification Authority. See also: [mutual authentication](#), [certificate](#).

certificate

A digital document used for security purposes that affirms a person or service's identity; issued by a Certificate Authority. See also: [Certificate Authority \(CA\)](#).

Certificate Authority (CA)

An entity which issues digital certificates for use by other parties to verify their credentials; also known as Certification Authority. See also: [mutual authentication](#), [certificate](#).

CA signing policy

A policy that places constraints on the information a given CA can bind to public keys; it constrains the identities a CA is trusted to assert in a certificate. See also: [Certificate Authority \(CA\)](#).

checkpointing

In Grid Engine, checkpointing is a procedure that saves the execution status of a job, allowing the job to be aborted and resumed later without loss of information or completed work. Checkpoints can be moved to another host before execution; this is called ‘migration’. See also: [migration](#).

cluster

A collection of machines, called hosts or nodes.

communication integrity

A phenomenon where an eavesdropper may be able to read communication between two parties, but due to security features, is not able to modify the communication in any way.

compute node

A UniCluster Express node running services that provide node monitoring capabilities and Grid Engine queue management. See also: [login node](#), [main node](#), [workstation node](#).

Condor

A open source job scheduler; for more information, see the Condor Project website, <http://www.cs.wisc.edu/condor/>. See also: [job scheduler](#).

container

A JSR168-compliant hosting environment running WSRF-compliant web services.

controllable ephemeral port

An ephemeral port that is constrained to a given port range. See also: [ephemeral port](#), [service port](#).

end-entity

Various entities (such as users and services) that use certificates to identify themselves.

endpoint reference (EPR)

An XML structure encapsulating information useful for addressing a message to a Web service; this includes the destination address of the message, any additional parameters (called reference parameters) necessary to route the message to the destination and optional metadata about the service.

ephemeral port

A non-deterministic port assigned by the system in the untrusted port range (>1024). See also: [controllable ephemeral port](#), [service port](#).

Enterprise Grid Forum

The Enterprise Grid Forum merged with the Global Grid Forum in 2006 to form the Open Grid Forum. See also: [Open Grid Forum](#).

EPR

See [endpoint reference \(EPR\)](#).

execution host

A Grid Engine host that has permission to execute jobs; execution hosts have queue instances attached to them and run the execution daemon sge_execd. See also: [administration host](#), [master host](#), [submit host](#).

fork

Term used to refer to a UNIX fork, supported Manage Job Service scheduler mechanism; used when no scheduler is installed.

Ganglia

An open source, scalable distributed monitoring system for high-performance computing systems such as clusters and grids. For more information, see the Ganglia website, <http://ganglia.sourceforge.net/>. See also: [gmetad](#), [gmond](#).

Global Grid Forum (GGF)

The community of users, developers and vendors leading the global standardization effort for grid computing. In 2006, the Global Grid Forum merged with the Enterprise Grid Alliance to form the Open Grid Forum. See also: [Open Grid Forum](#).

Globus Toolkit

An open source software toolkit used for building grid systems and applications; for more information, refer to the Globus Toolkit website, <http://www.globus.org/>.

gmetad

A multi-threaded Ganglia daemon which runs on each cluster node that is monitored.

gmond

A Ganglia daemon that periodically polls a collection of child data sources, parses the collected data, saves all numeric, volatile metrics to a round-robin database, and exports the aggregated data to clients. See also: [gmetad](#).

Grid Engine

An open source batch-queuing system that is typically used on a computer cluster. Grid Engine accepts, schedules, dispatches, and manages the remote execution of large numbers of standalone, parallel, or interactive user jobs. For more information, see the Grid Engine website, <http://gridengine.sunsource.net/>.

gridmap-file

A file containing entries mapping certificate subjects to local usernames; some Globus Toolkit installations use gridmap-files to identify users.

Grid File Transfer Protocol (GridFTP)

A high-performance, secure, reliable data transfer protocol optimized for high-bandwidth wide-area networks. See also: [Web Services Grid Resource Allocation and Management \(WS-GRAM\)](#).

Grid Security Infrastructure (GSI)

A specification for secret, tamper-proof, delegatable communication between software in the grid computing environment.

GSI-protected connection

A network connection authenticated GSI; these connections are integrity protected and can be encrypted if so requested by the controlling application. See also: [Grid Security Infrastructure \(GSI\)](#).

GSI-OpenSSH

An SSH server that integrates with the GSI authorization framework used in UniCluster Express. See also: [Grid Security Infrastructure \(GSI\)](#)

hard resource requirement

In Grid Engine, hard resource requirements are resources that must be allocated before a job can start. See also: [soft resource requirement](#).

hosting environment

A type of container encapsulating a number of different services and acting as a single network endpoint for those services.

IANA

See [Internet Assigned Numbers Authority \(IANA\)](#).

interactive job

A Grid Engine session started with the commands `qrsh`, `qsh`, or `qlogin`, which open an Xterm window for the user. See also: [batch job](#).

Internet Assigned Numbers Authority (IANA)

An organization dedicated to preserving the central coordinating functions of the global Internet for the public good; see the IANA website, <http://www.iana.org/>.

IPC channel

The connection between a GridFTP front-end process and back-end data node. See also: [striped configuration](#).

Java Naming and Directory Interface (JNDI)

A type of file used for Web service configuration files. See also: [Web Service Deployment Descriptor \(WSDD\)](#).

Java Runtime Environment (JRE)

The JRE is a Java Virtual Machine bundled along with a set of standard class libraries which implement the Java API (Application Programming Interface). See also: [Java Virtual Machine \(JVM\)](#).

Java Virtual Machine (JVM)

A set of computer software programs and data structures which implements a specific virtual machine model.

JNDI

See [Java Naming and Directory Interface \(JNDI\)](#).

job

A unit of work; jobs are scripts or executables that are normally run from a command line and have resources associated with them. Jobs usually run in batch, rather than interactive, mode.

job scheduler

Software application that is in charge of unattended background executions and batch workload processing.

JRE

See [Java Runtime Environment \(JRE\)](#).

JVM

See [Java Virtual Machine \(JVM\)](#).

Load Sharing Facility (LSF)

A proprietary job scheduler. See also: [job scheduler](#).

Load Leveler

A proprietary job scheduler. See also: [job scheduler](#).

Log4J

A Java package that enables logging at runtime without modifying the application binary; for more information, see the Apache log4j web page, <http://logging.apache.org/log4j/docs/>.

login node

A UniCluster Express node running services that allow a user to log into the cluster, provides node monitoring capabilities, allows users to submit Grid Engine jobs, and provides Grid Engine queue management. See also: [compute node](#), [main node](#), [workstation node](#).

LSF

See [Load Sharing Facility \(LSF\)](#).

main node

A UniCluster Express node running all services, including cluster security. All other nodes can be referred to as ‘subordinate’ nodes. See also: [compute node](#), [login node](#), [workstation node](#).

master host

A Grid Engine host that is central to the overall cluster activity. The master runs the master daemon `sge_qmaster`, and the scheduler daemon `sge_schedd`. These daemons control all Grid Engine system components. By default, the master host is also an administration and submit host. See also: [administration host](#), [execution host](#), [submit host](#).

migration

The process in Grid Engine of moving a job checkpoint to another host before resuming job execution. See also: [checkpointing](#).

Monitoring and Discovery Service (MDS)

A Globus Toolkit information services component that provides information about available resources and their status.

mutual authentication

The process of two parties authenticating each other using certificates and public keys in such a way that both parties are assured of the others's identity; also called two-way authentication. See also: [certificate](#).

Network File System (NFS)

A method of creating file systems on diskless clients by providing remote access to shared file systems across networks.

OASIS

See [Organization for the Advancement of Structured Information Standards \(OASIS\)](#).

OGSA

See [Open Grid Services Architecture \(OGSA\)](#).

OGSI

See [Open Grid Services Infrastructure \(OGSI\)](#).

Open Grid Forum

The Open Grid Forum was formed in 2006, when the Global Grid Forum and Enterprise Grid Alliance merged. This group is an international community dedicated to accelerating grid adoption by providing an open forum for grid innovation and developing open standards for grid software interoperability.

Open Grid Services Architecture (OGSA)

An architecture for a service-oriented grid computing environment for business and scientific use.

Open Grid Services Infrastructure (OGSI)

A Web service that defines mechanisms for creating, managing and exchanging information among grid services; for more information, see 'Open Grid Services Infrastructure', <http://xml.coverpages.org/OGSI-SpecificationV110.pdf>.

OpenSSL

An open source Secure Sockets Layer implementation used by GSI; for more information, see the OpenSSL Project website, <http://www.openssl.org>. See also: [GSI-OpenSSH](#), [Secure Sockets Layer/Transport Layer Security \(SSL/TLS\)](#).

Organization for the Advancement of Structured Information Standards (OASIS)

A global consortium that drives the development, convergence and adoption of e-business and Web service standards; for more information, see the OASIS website, <http://www.oasis-open.org/home/index.php>.

parallel environment

A system configuration that defines the necessary interfaces for Grid Engine to correctly handle parallel jobs. See also: [parallel job](#).

parallel job

A Grid Engine job made up of more than one closely correlated task; tasks can be distributed across multiple hosts. Parallel jobs usually use communication tools such as shared memory or message passing to synchronize and correlate tasks. See also: [parallel environment](#).

PBS

See [Portable Batch Systems \(PBS\)](#).

PKI

See [Public Key Infrastructure \(PKI\)](#).

policy

A set of rules and configurations that defines the behavior of the Grid Engine system; policies are implemented automatically.

Portable Batch Systems (PBS)

An open source job scheduler; for more information, see the OpenPBS Project website, <http://www.openpbs.org>. See also: [job scheduler](#).

PostgreSQL

An open source, object-relational database management system (ORDBMS); for more information, refer to the PostgreSQL website, <http://www.postgresql.org/>.

Pre-WS

Globus Toolkit services which predate Web services. See also: [Web Services Resource Framework \(WSRF\)](#).

proxy certificate

A short-lived certificate issued by a user and based on a certificate issued by a Certificate Authority. See also: [certificate](#), [Certificate Authority \(CA\)](#), [mutual authentication](#).

Public Key Infrastructure (PKI)

A combination of software, hardware and procedures that allow users and services to authenticate one other using identity certificates and also allows users to encrypt and decrypt messages travelling over the network. See also: [asymmetric cryptography](#).

queue

A Grid Engine ‘queue’ is container for a class of job that are allowed to run concurrently. A queue should be thought of as ‘a place where a job runs’, rather than ‘a place for submitted jobs to wait’. Throughout its lifetime, a running job is associated with its queue and is affected by things that happen to a job; for example, if a queue is suspended, all jobs associated with that queue are also suspended.

RFT

See [Reliable File Transfer Service \(RFT\)](#).

Replica Location Service (RLS)

A Globus Toolkit service that maintains and provides access to the mapping information from logical names for data items to target names.

Reliable File Transfer Service (RFT)

A Globus Toolkit service that provides job scheduler-like functionality for data movement. See also: [Grid File Transfer Protocol \(GridFTP\)](#), [Web Services Resource Framework \(WSRF\)](#).

resource

In Globus Toolkit, an entity that contains state information about a Web service. In Grid Engine, a computational device consumed or occupied by a running job; examples are: memory, CPU, I/O bandwidth, file space.

resource properties

In the Globus Toolkit, a resource property is a piece of information defining part of a resource; a resource property may reflect the resource’s state, metadata, manageability information, etc.

RLS

See [Replica Location Service \(RLS\)](#).

RRDTool

An open source, industry standard, high performance data logging and graphing tool for time series data; for details, refer to the RRDTool website, <http://oss.oetiker.ch/rrdtool/>.

SAAJ

See [SOAP with Attachments for Java \(SAAJ\)](#).

SASL

See [Simple Authentication and Security Layer \(SASL\)](#).

SEG

See [Scheduler Event Generator \(SEG\)](#).

Scheduler Event Generator (SEG)

A program which uses scheduler-specific monitoring modules to generate job state change events.

sge_execd

The Grid Engine execution daemon responsible for queue instances on its host and for running jobs in the queue instances. See also: [sge_schedd](#), [sge_qmaster](#).

sge_schedd

The Grid Engine scheduling daemon. See also: [sge_execd](#), [sge_qmaster](#).

sge_qmaster

The Grid Engine daemon that maintains tables about hosts, queues, jobs, system load, and user permissions; sge_qmaster is only run on the UniCluster Express main node. See also: [sge_execd](#), [sge_schedd](#).

slots

In Grid Engine, the ‘slots’ is the number of jobs that are allowed to run concurrently in the queue. Slots may also be referred to as ‘job slots’.

SOAP with Attachments for Java (SAAJ)

A standard to send XML documents over the Internet from the Java platform; for more information, see the Sun Developer Network SAAJ web page, <http://java.sun.com/webservices/saaj/index.jsp>.

Simple Authentication and Security Layer (SASL)

An open source package used by GSI; for more information, see the Carnegie Mellon University SASL web page, <http://asg.web.cmu.edu/sasl>.

Secure Sockets Layer/Transport Layer Security (SSL/TLS)

Cryptographic protocols which provides secure communications on the Internet.

service port

Static ports for well-known Globus Toolkit services. See also: [ephemeral port](#).

SimpleCA

A Globus Toolkit package that provides a simplified certification authority to users and services.

soft resource requirement

In Grid Engine, soft resource requirements are resources that a job needs, but does not have to be allocated, before the job starts. See also: [hard resource requirement](#).

SSL/TLS

See [Secure Sockets Layer/Transport Layer Security \(SSL/TLS\)](#).

standalone container

A simple HTTP server that passes requests to the SOAP engine; a standalone container can only serve .wsdl and .xsd files.

striped configuration

A GridFTP server configuration in which a single front-end process has several back-end data nodes available. See also: [IPC channel](#).

submit host

A Grid Engine host that allows users to submit and control batch jobs. See also: [administration host](#), [execution host](#), [master host](#).

subordinate node

A UniCluster Express node other than the main node. See also: [main node](#), [compute node](#), [login node](#), [workstation node](#).

Tomcat

A type of container; for more information, see the Apache Tomcat Project web page, <http://tomcat.apache.org/>.

Torque

An open source job scheduler; for more information, see the Cluster Resources website, <http://www.clusterresources.com/pages/products/torque-resource-manager.php>. See also: [job scheduler](#).

Web Services Agreement (WS-Agreement)

A Web services protocol for establishing agreements between service providers and service consumers.

Web Services Core (WS Core)

A set of libraries and tools that allows Web services to be platform independent.

Web Services Description Language (WSDL)

An XML format published for describing Web services.

Web Services Description Language for Java Toolkit (WSDL4J)

A WSDL for Java that allows the creation, representation and manipulation of WSDL documents. See also: [Web Services Description Language \(WSDL\)](#).

Web Service Deployment Descriptor (WSDD)

A type of file used for Web service configuration files. See also: [Java Naming and Directory Interface \(JNDI\)](#).

Web Services Distributed Management (WSDM)

A model for Web services management that allows interoperability between multiple environments.

Web Services Grid Resource Allocation and Management (WS-GRAM)

A Globus Toolkit set of services and clients for communicating with a range of different batch/cluster job schedulers using a common protocol.

Web Services Invocation Framework (WSIF)

A simple Java API for invoking Web services; for more information, see the Apache WSIF web page, <http://ws.apache.org/wsif/>.

Web Services Resource (WS-Resource)

A family of specifications that enable and standardize interfaces for Web services. See also: [Web Services Resource Framework \(WSRF\)](#).

Web Services Resource Framework (WSRF)

A specification that extends Web services for grid applications by giving them the ability to retain state information while at the same time retaining statelessness. The combination of a Web service and resource is referred to as a WS-Resource; the WSRF is a collection of different specifications that manage WS-Resources. For more information, see the Globus WS-Resource Framework web page, <http://www.globus.org/wsrf/>. See also: [Web Services Resource \(WS-Resource\)](#).

well known port

A port number registered with IANA. See also: [Internet Assigned Numbers Authority \(IANA\)](#), [ephemeral port](#), [service port](#).

workstation node

A UniCluster Express node running services allows a user to log in to the cluster and provides node monitoring capabilities. See also: [compute node](#), [login node](#), [main node](#).

WSDD

See [Web Service Deployment Descriptor \(WSDD\)](#).

WSDL4J

See [Web Services Description Language for Java Toolkit \(WSDL4J\)](#).

WSDM

See [Web Services Distributed Management \(WSDM\)](#).

WS-GRAM

See [Web Services Grid Resource Allocation and Management \(WS-GRAM\)](#).

WSIF

See [Web Services Invocation Framework \(WSIF\)](#).

WSRF

See [Web Services Resource Framework \(WSRF\)](#).

Index

B

[bookmark](#), 30

D

[deleting files using RFT](#), 55
[deleting resource property](#), 71
[destroying resources](#), 61

G

[Ganglia](#), 9
 [gemtric](#), 9
 [gstat](#), 10
[getting values of multiple resource properties](#), 67
[getting values of resource property](#), 65
[globusrun-ws](#), 43
[globus-url-copy](#), 39
[grid engine](#), 5
 [check status](#), 6
 [delete job](#), 6
 [queue information](#), 7
 [submit job](#), 5
[GSI-OpenSSH](#)
 [gsiscp](#), 49
 [gsissh](#), 49

I

[inserting values to resource property](#), 68

L

[log in Cluster](#), 1

M

Monitor Console, 13

- changing preferences, 17
- changing time ranges, 20
- cluster node details, 24
- cluster node status, 23
- cluster nodes summary, 21
- filtering information, 29
- host details, 32
- job details, 38
- job queues, 36
- quitting, 38
- starting console, 13
- using cluster workstation, 13
- using non-cluster machine, 15

P

performing query on resource property, 64

R

Reliable File Transfer, 53

S

setting termination time, 62

U

unicluster-grid-logon, 2

updating values of resource property, 70

Using rft, 54

W

WSN command line tools, 73

- getting current message, 73
- pausing subscriptions, 74
- resuming subscriptions, 75
- subscribing specific topics, 76