



NORDUGRID

NORDUGRID-MANUAL-1
07/20/2006

The ARC user interface User's manual

Mattias Ellert

Abstract

This is the user's manual to the ARC user interface.

Table of Contents

1 The ARC user interface commands.....	4
1.1 ngsb.....	4
1.2 ngstat.....	5
1.3 ngcat.....	6
1.4 ngget.....	7
1.5 ngkill.....	8
1.6 ngresub.....	9
1.7 ngclean.....	11
1.8 ngrenew.....	12
1.9 ngsync.....	12
1.10 ngcopy.....	13
1.11 ngremove.....	14
1.12 ngls.....	14
1.13 ngacl.....	15
1.14 ngrequest.....	15
2 User configuration files.....	16
2.1 .ngalias.....	16
2.2 .nggiislist.....	16
2.3 .nghistory.....	16
2.4 .ngjobs.....	16
2.5 .ngrc.....	16

1 The ARC user interface commands

The command line user interface of ARC consists of a set of commands necessary for job submission and manipulation and data management. These are described in the following sections.

1.1 *ngsub*

The *ngsub* command is used for submitting jobs to the NorduGrid. The jobs are described using the extended resource description language (xrsl).

Usage: *ngsub* [options] [task ...]

Options:

-c, -cluster	<i>[-]name</i>	explicitly select or reject a specific cluster
-C, -clustlist	<i>[-]filename</i>	list of clusters to select or reject
-g, -giisurl	<i>URL</i>	URL to a central GIIS
-G, -giislist	<i>filename</i>	list of GIIS URLs
-f, -file	<i>filename</i>	xrsl-file describing the job to be submitted ¹
-e, -xrsl	<i>xrsl string</i>	xrsl string describing the job to be submitted ²
-o, -joblist	<i>filename</i>	file where the jobIDs will be stored
-dryrun		add dryrun option to the xrsl
-dumpxrsl		do not submit – dump transformed xrsl to stdout
-t, -timeout	<i>time</i>	timeout for MDS queries in seconds (default 40)
-d, -debug	<i>debuglevel</i>	0 = none, 1 = some, 2 = more, 3 = a lot
-x, -anonymous		use anonymous bind for MDS queries
-X, -gsi		use gsi-gssapi bind for MDS queries
-v, -version		print version information
-h, -help		print this help

Arguments:

task ...	xrsl strings ¹ or xrsl-files ² describing the jobs to be submitted
----------	--

The *ngsub* command locates the available clusters by querying a Grid information index server (GIIS). Which GIIS to query can be specified by giving the *-giisurl* option to the command. The URL to the GIIS should be given in the format

`ldap://hostname[:port]/basedn`

If the port number is omitted the default 2135 is used. Two different syntax for the base DN of the LDAP query are accepted, e.g. a NorduGrid top-level GIIS can be specified in either of the two following ways:

```
ldap://index1.nordugrid.org:2135/O=Grid/mds-vo-name=NorduGrid
ldap://index1.nordugrid.org:2135/mds-vo-name=NorduGrid,O=Grid
```

Several GIISes can be specified by repeating the *-giisurl* option:

¹ NorduGrid/ARC releases before 0.5.29

² NorduGrid/ARC releases 0.5.29 and above

```
ngsub -giisurl url1 -giisurl url2 ...
```

The GIIS URLs can also be read from a file if the `-giislist` option is used. If neither the `-giisurl` option nor the `-giislist` option is given, a list of URLs is read from a default `giislist` file. The first existing file in the following list is used:

1. `${HOME}/.nggiislist`
2. `${NORDUGRID_LOCATION}/etc/giislist`
or `/opt/nordugrid/etc/giislist` if `${NORDUGRID_LOCATION}` is undefined
3. `/etc/giislist`

The `-cluster` option can be used to force the job to be submitted to a particular cluster, or to reject submission to a particular cluster. If the `-clustlist` option is used the names of the clusters to select or reject are read from a file. The `-cluster` and `-clustlist` options can be repeated several times.

Examples:

```
ngsub -c Grid.nbi.dk -c Grid.tsl.uu.se ...  
submit the job to either Grid.nbi.dk or Grid.tsl.uu.se
```

```
ngsub -c -quark.hep.lu.se -c -Grid.nbi.dk ...  
don't submit the job to quark.hep.lu.se or Grid.nbi.dk
```

The `xrsl` that describes the job can be given either as an argument on the command line or can be read from a file by using the `-file` option¹. Several jobs can be requested at the same time by giving more than one `xrsl` argument or by repeating the `-file` (or `-xrsl`) option. It is also possible to mix `xrsl` arguments and `-file` (`-xrsl`) options in the same `ngsub` command.

The user interface transforms the `xrsl` into a format that can be understood by the Grid Manager to which it is being submitted. By specifying the `-dumpxrsl` option the transformed `xrsl` is written to stdout instead of being submitted to the remote cluster.

If the job is successfully submitted, a `jobID` is printed to standard output. This `jobID` is a URL that uniquely identifies the job while it is in the system. If the `-joblist` option is given, the `jobID` is also written to a file with the specified file name. This file can later be used with the `-joblist` option of the other user interface commands.

For setting defaults, see the section about `.ngrc` (section 2.5).

1.2 *ngstat*

The `ngstat` command is used for obtaining the status of jobs that have been submitted to NorduGrid and the status of the clusters in the Grid.

Usage: `ngstat [options] [job ...]`

¹ Starting with release 0.5.29, the logics is reverted: the command line argument is a file, while `xrsl` string should be indicated with the `-xrsl` option

Options:

-a, -all		all jobs
-i, -joblist	<i>filename</i>	file containing a list of jobIDs
-c, -cluster	<i>[-]name</i>	explicitly select or reject a specific cluster
-C, -clustlist	<i>[-]filename</i>	list of clusters to select or reject
-s, -status	<i>statusstr</i>	only select jobs whose status is <i>statusstr</i>
-g, -giisurl	<i>URL</i>	URL to a central GIIS
-G, -giislist	<i>filename</i>	list of GIIS URLs
-q, -queues		show information about clusters and queues
-l, -long		long format (more information)
-t, -timeout	<i>time</i>	timeout for MDS queries in seconds (default 40)
-d, -debug	<i>debuglevel</i>	0 = none, 1 = some, 2 = more, 3 = a lot
-x, -anonymous		use anonymous bind for MDS queries
-X, -gsi		use gsi-gssapi bind for MDS queries
-v, -version		print version information
-h, -help		print this help

Arguments:

job ... list of jobIDs and/or jobnames

The `ngstat` command gives the status of a job submitted to NorduGrid. The job can be referred to either by the `jobID` that was returned by `ngsub` at submission time or by its name if the `xrsl` that was submitted containing a `jobname` attribute.

More than one `jobID` and/or job name can be given. If several jobs were submitted with the same job name, the status of all those jobs are shown. If the `-joblist` option is used, the list of `jobIDs` is read from a file with the specified name. By specifying the `-all` option, the status of all jobs in the system will be shown. If the `-long` option is given, more detailed information is displayed.

The `-cluster` and `-clustlist` options can be used to select or reject jobs at specific clusters. The `-status` option can be used to select jobs in a specific state. For finished jobs, the `-status` option will match `FINISHED` or `FAILED` depending on whether the job finished successfully or not. These options can be repeated several times.

If the `-queues` option is used, `ngstat` gives the status of the clusters and queues available rather than of the jobs submitted. In this case the `-cluster` and `-clustlist` options can be used to select or reject clusters for which the status should be returned. For a description of the `-giisurl` and `-giislist` options refer to the `ngsub` section (section 1.1). Also in this case the `-long` option can be used to obtain more detailed information.

For setting defaults, see the section about `.ngrc` (section 2.5).

1.3 `ngcat`

The `ngcat` command can be used to view the standard output, standard error, and Grid Manager log records of a job.

Usage: `ngcat [options] [job ...]`

Options:

<code>-a, -all</code>		all jobs
<code>-i, -joblist</code>	<i>filename</i>	file containing a list of jobIDs
<code>-c, -cluster</code>	<code>[-]name</code>	explicitly select or reject a specific cluster
<code>-C, -clustlist</code>	<code>[-]filename</code>	list of clusters to select or reject
<code>-s, -status</code>	<i>statusstr</i>	only select jobs whose status is <i>statusstr</i>
<code>-o, -stdout</code>		show the stdout of the job (default)
<code>-e, -stderr</code>		show the stderr of the job
<code>-l, -gmlog</code>		show the Grid manager's error log of the job
<code>-t, -timeout</code>	<i>time</i>	timeout for MDS queries in seconds (default 40)
<code>-d, -debug</code>	<i>debuglevel</i>	0 = none, 1 = some, 2 = more, 3 = a lot
<code>-x, -anonymous</code>		use anonymous bind for MDS queries
<code>-X, -gsi</code>		use gsi-gssapi bind for MDS queries
<code>-v, -version</code>		print version information
<code>-h, -help</code>		print this help

Arguments:

`job ...` list of jobIDs and/or jobnames

The `ngcat` command displays the standard output, standard error or Grid Manager log records of jobs. The job can be referred to either by the `jobID` that was returned by `ngsub` at submission time, or by its name if the `xrsl` that was submitted containing a `jobname` attribute.

More than one `jobID` and/or job name can be given. If several jobs were submitted with the same job name, the standard output, standard error or Grid Manager log of all those jobs are shown. If the `-joblist` option is used, the list of `jobIDs` is read from a file with the specified name. By specifying the `-all` option, the standard output, standard error or Grid Manager log of all the jobs in the system will be shown.

The `-cluster` and `-clustlist` options can be used to select or reject jobs at specific clusters. The `-status` option can be used to select jobs in a specific state. For finished jobs, the `-status` option will match `FINISHED` or `FAILED` depending on whether the job finished successfully or not. These options can be repeated several times.

Only jobs where the `stdout`, `stderr` and `gmlog` arguments were given in the `xrsl` can display the contents of those files.

For setting defaults, see the section about `.ngrc` (section 2.5).

1.4 `ngget`

The `ngget` command is used for retrieving the results from a job.

Usage: `ngget [options] [job ...]`

Options:

-a, -all		all jobs
-i, -joblist	<i>filename</i>	file containing a list of jobIDs
-c, -cluster	<i>[-]name</i>	explicitly select or reject a specific cluster
-C, -clustlist	<i>[-]filename</i>	list of clusters to select or reject
-s, -status	<i>statusstr</i>	only select jobs whose status is <i>statusstr</i>
-dir	<i>dirname</i>	download directory (the job directory will be created in this directory)
-j, -usejobname		use the job name instead of the short ID as the job directory name
-keep		keep files on gatekeeper (do not clean)
-t, -timeout	<i>time</i>	timeout for MDS queries in seconds (default 40)
-d, -debug	<i>debuglevel</i>	0 = none, 1 = some, 2 = more, 3 = a lot
-x, -anonymous		use anonymous bind for MDS queries
-X, -gsi		use gsi-gssapi bind for MDS queries
-v, -version		print version information
-h, -help		print this help

Arguments:

job ... list of jobIDs and/or jobnames

This command downloads the results after a job has completed on the NorduGrid. Only the results of jobs that have finished can be downloaded. The job can be referred to either by the jobID that was returned by ngsuB at submission time or by its name if the xrsL that was submitted contained a jobname attribute.

More than one jobID and/or job name can be given. If several jobs were submitted with the same name, the results of all those jobs are downloaded. If the -joblist option is used, the list of jobIDs is read from a file with the specified name. By specifying the -all option, the results of all jobs in the system are downloaded.

The -cluster and -clustlist options can be used to select or reject jobs at specific clusters. The -status option can be used to select jobs in a specific state. For finished jobs the -status option will match FINISHED or FAILED depending on whether the job finished successfully or not. These options can be repeated several times.

For each job that is downloaded, a subdirectory will be created in the current or a dedicated/specified download directory, that will contain the downloaded files.

If the download was successful, the job will be removed from the remote cluster, unless the -keep option was specified.

For setting defaults, see the section about .ngrc (section 2.5).

1.5 ngkill

The ngkill command is used to kill running jobs in the NorduGrid.

Usage: `ngkill [options] [job ...]`

Options:

<code>-a, -all</code>		all jobs
<code>-i, -joblist</code>	<i>filename</i>	file containing a list of jobIDs
<code>-c, -cluster</code>	<code>[-]name</code>	explicitly select or reject a specific cluster
<code>-C, -clustlist</code>	<code>[-]filename</code>	list of clusters to select or reject
<code>-s, -status</code>	<i>statusstr</i>	only select jobs whose status is <i>statusstr</i>
<code>-keep</code>		keep files on gatekeeper (do not clean)
<code>-t, -timeout</code>	<i>time</i>	timeout for MDS queries in seconds (default 40)
<code>-d, -debug</code>	<i>debuglevel</i>	0 = none, 1 = some, 2 = more, 3 = a lot
<code>-x, -anonymous</code>		use anonymous bind for MDS queries
<code>-X, -gsi</code>		use gsi-gssapi bind for MDS queries
<code>-v, -version</code>		print version information
<code>-h, -help</code>		print this help

Arguments:

<code>job ...</code>	list of jobIDs and/or jobnames
----------------------	--------------------------------

This command kills a running job on the NorduGrid. The job can be referred to either by the jobID that was returned by `ngsub` at submission time, or by its job name if the `xrsl` that was submitted containing a `jobname` attribute.

More than one jobID and/or job name can be given. If several jobs were submitted with the same name, **all those jobs are killed**. If the `-joblist` option is used, the list of jobIDs is read from a file with the specified name. By specifying the `-all` option, all jobs can be killed.

The `-cluster` and `-clustlist` options can be used to select or reject jobs at specific clusters. The `-status` option can be used to select jobs in a specific state. For finished jobs, the `-status` option will match FINISHED or FAILED depending on whether the job finished successfully or not. These options can be repeated several times.

If the job is successfully killed, the job is removed from the remote cluster, unless the `-keep` option was specified.

For setting defaults, see the section about `.ngrc` (section 2.5).

1.6 `ngresub`

The `ngresub` command is used for resubmitting jobs on the NorduGrid.

Usage: `ngresub [options] [job ...]`

Options:

<code>-a, -all</code>		all jobs
<code>-i, -joblist</code>	<i>filename</i>	file containing a list of jobIDs
<code>-c, -cluster</code>	<code>[-]name</code>	explicitly select or reject a specific cluster
<code>-C, -clustlist</code>	<code>[-]filename</code>	list of clusters to select or reject

Options:

<code>-s, -status</code>	<i>statusstr</i>	only select jobs whose status is <i>statusstr</i>
<code>-k, -kluster</code>	<code>[-]name</code>	explicitly select or reject a specific cluster as resubmission target
<code>-K, -klustlist</code>	<code>[-]filename</code>	list of clusters to select or reject as resubmission target
<code>-g, -giisurl</code>	<i>URL</i>	URL to a central GIIS
<code>-G, -giislist</code>	<i>filename</i>	list of GIIS URLs
<code>-o, -joblist</code>	<i>filename</i>	file where the jobIDs will be stored
<code>-dryrun</code>		add dryrun option to the xrsl
<code>-dumpxrsl</code>		do not submit – dump transformed xrsl to stdout
<code>-keep</code>		keep files on gatekeeper (do not clean)
<code>-t, -timeout</code>	<i>time</i>	timeout for MDS queries in seconds (default 40)
<code>-d, -debug</code>	<i>debuglevel</i>	0 = none, 1 = some, 2 = more, 3 = a lot
<code>-x, -anonymous</code>		use anonymous bind for MDS queries
<code>-X, -gsi</code>		use gsi-gssapi bind for MDS queries
<code>-v, -version</code>		print version information
<code>-h, -help</code>		print this help

Arguments:

`job ...` list of jobIDs and/or jobnames

The `ngresub` command re-submits a previously submitted job to the NorduGrid. If the xrsl describing the job contained input file locations specified as relative paths, the command must be issued in the same working directory as the original `ngsub` submission.

More than one jobID and/or job name can be given. If several jobs were submitted with the same name, all those jobs will be re-submitted. If the `-joblist` option is used, the list of jobIDs is read from a file with the specified name. By specifying the `-all` option, all jobs can be resubmitted.

The `-cluster` and `-clustlist` options can be used to select or reject jobs at specific clusters. The `-status` option can be used to select jobs in a specific state. For finished jobs, the `-status` option will match FINISHED or FAILED depending on whether the job finished successfully or not. These options can be repeated several times.

The `-kluster` and `-klustlist` options, on the other hand, are used to define the clusters to which the **re-submitted** jobs will be sent. The cluster to which the job was previously submitted will be excluded from the list of possible destinations for resubmission. Also these options can be repeated several times. For a description of the `-giisurl` and `-giislist` options, see the `ngsub` section (section 1.1).

If the re-submission was successful, the old job will be removed from the remote cluster, unless the `-keep` option was specified. Old Grid Manager log records are not preserved and are not transferred to the new site..

Only jobs where the `gmlog` argument was given in the xrsl can be resubmitted.

For setting defaults, see the section about `.ngrc` (section 2.5).

1.7 *ngclean*

The `ngclean` command removes the job from the remote cluster.

Usage: `ngclean [options] [job ...]`

Options:

<code>-a, -all</code>		all jobs
<code>-i, -joblist</code>	<i>filename</i>	file containing a list of jobIDs
<code>-c, -cluster</code>	<code>[-]name</code>	explicitly select or reject a specific cluster
<code>-C, -clustlist</code>	<code>[-]filename</code>	list of clusters to select or reject
<code>-s, -status</code>	<i>statusstr</i>	only select jobs whose status is <i>statusstr</i>
<code>-f, -force</code>		removes the job from the local list of jobs even if the job is not found in the MDS
<code>-t, -timeout</code>	<i>time</i>	timeout for MDS queries in seconds (default 40)
<code>-d, -debug</code>	<i>debuglevel</i>	0 = none, 1 = some, 2 = more, 3 = a lot
<code>-x, -anonymous</code>		use anonymous bind for MDS queries
<code>-X, -gsi</code>		use gsi-gssapi bind for MDS queries
<code>-v, -version</code>		print version information
<code>-h, -help</code>		print this help

Arguments:

<code>job ...</code>	list of jobIDs and/or jobnames
----------------------	--------------------------------

The `ngclean` command removes a job from the remote cluster. Only jobs that have finished can be removed. The job can be referred to either by the `jobID` that was returned by `ngsub` at submission time, or by its job name if the `xrsl` that was submitted containing a `jobname` attribute.

More than one `jobID` and/or job name can be given. If several jobs were submitted with the same name, all those jobs are removed. If the `-joblist` option is used, the list of `jobIDs` is read from a file with the specified `filename`. By specifying the `-all` option, all finished jobs can be removed.

The `-cluster` and `-clustlist` options can be used to select or reject jobs at specific clusters. The `-status` option can be used to select jobs in a specific state. For finished jobs, the `-status` option will match `FINISHED` or `FAILED` depending on whether the job finished successfully or not. These options can be repeated several times.

The `-force` option removes the job from your local list of jobs (see section 2.4), even if the job can not be found in the information system.

For setting defaults, see the section about `.ngrc` (section 2.5).

1.8 ngrenew

The `ngrenew` command can be used to renew the proxies of NorduGrid jobs.

Usage: `ngrenew [options] [job ...]`

Options:

<code>-a, -all</code>		all jobs
<code>-i, -joblist</code>	<i>filename</i>	file containing a list of jobIDs
<code>-c, -cluster</code>	<code>[-]name</code>	explicitly select or reject a specific cluster
<code>-C, -clustlist</code>	<code>[-]filename</code>	list of clusters to select or reject
<code>-s, -status</code>	<i>statusstr</i>	only select jobs whose status is <i>statusstr</i>
<code>-t, -timeout</code>	<i>time</i>	timeout for MDS queries in seconds (default 40)
<code>-d, -debug</code>	<i>debuglevel</i>	0 = none, 1 = some, 2 = more, 3 = a lot
<code>-x, -anonymous</code>		use anonymous bind for MDS queries
<code>-X, -gsi</code>		use gsi-gssapi bind for MDS queries
<code>-v, -version</code>		print version information
<code>-h, -help</code>		print this help

Arguments:

<code>job ...</code>	list of jobIDs and/or jobnames
----------------------	--------------------------------

This command renews the proxy of a job on the NorduGrid. The job can be referred to either by the `jobID` that was returned by `ngsub` at submission time or by its name if the `xrsl` that was submitted contained a `jobname` attribute.

More than one `jobID` and/or job name can be given. If several jobs were submitted with the same name, the proxies of all those jobs are renewed. If the `-joblist` option is used, the list of `jobIDs` is read from a file with the specified filename. By specifying the `-all` option, the proxies of all jobs can be renewed.

The `-cluster` and `-clustlist` options can be used to select or reject jobs at specific clusters. The `-status` option can be used to select jobs in a specific state. For finished jobs, the `-status` option will match `FINISHED` or `FAILED` depending on whether the job finished successfully or not. These options can be repeated several times.

For setting defaults, see the section about `.ngrc` (section 2.5).

1.9 ngsync

The `ngsync` command synchronizes your local list of jobs with the information in the information system. Run this command every time you switch workstation.

Usage: `ngsync [options]`

Options:

<code>-c, -cluster</code>	<code>[-]name</code>	explicitly select or reject a specific cluster
<code>-C, -clustlist</code>	<code>[-]filename</code>	list of clusters to select or reject
<code>-g, -giisurl</code>	<i>URL</i>	URL to a central GIIS

Options:

-G, -giislist	<i>filename</i>	list of GIIS URLs
-f, -force		don't ask for verification
-t, -timeout	<i>time</i>	timeout for MDS queries in seconds (default 40)
-d, -debug	<i>debuglevel</i>	0 = none, 1 = some, 2 = more, 3 = a lot
-x, -anonymous		use anonymous bind for MDS queries
-X, -gsi		use gsi-gssapi bind for MDS queries
-v, -version		print version information
-h, -help		print this help

The ARC user interface keeps a local list of active jobs in the user's home directory (see section 2.4). If this file is lost, or the user wants to recreate the file on a different workstation, the `ngsync` command can be used to recreate the file from the information available in the information system.

Since the information about a job in the system can be slightly out of date in case the user submitted or removed a job very recently, a warning is issued when this command is run. The `-force` option disables this warning.

The `-cluster` and `-clustlist` options can be used to select or reject clusters that should be considered in the synchronization. These options can be repeated several times. For a description of the `-giisurl` and `-giislist` options the `ngsub` section (section 1.1)

For setting defaults, see the section about `.ngrc` (section 2.5).

1.10 *ngcopy*¹

The `ngcopy` command can be used to copy data between various data servers.

Usage: `ngcopy [options] source destination`

Options:

-p, -passive		use passive transfer (does not work if secure is on, default if secure is not requested)
-n, -nopassive		do not try to force passive transfers
-T, -notransfer		do not transfer file, just register it - destination must be non-existing meta-URL.
-u, -secure		use secure transfer (insecure by default)
-y, -cache	<i>path</i>	path to local cache (use to put file into cache)
-Y, -cachedata	<i>path</i>	path for cache data (if different from -y)
-t, -timeout	<i>time</i>	timeout for MDS queries in seconds (default 40)
-d, -debug	<i>debuglevel</i>	-3 = quiet, 3 = a lot
-i, -indicate		show progress bar for debug levels -2 and lower
-v, -version		print version information
-h, -help		print this help

¹ Called **ngcp** in releases 0.5.29 and above

Arguments:

source	<i>URL</i>	source URL
destination	<i>URL</i>	Destination URL

This command transfers contents of a file between 2 end-points. End-points are represented by URLs or meta-URLs. For supported endpoints please read "Protocols, Uniform Resource Locators (URL) And Extensions Supported in ARC".

For setting defaults, see the section about `.ngrc` (section 2.5).

1.11 *ngremove*¹

The `ngremove` command can be used to remove stored objects.

Usage: `ngremove [options] URL`

Options:

<code>-f, -force</code>		force removal of LFN even if not all physical instances were removed
<code>-t, -timeout</code>	<i>time</i>	timeout for MDS queries in seconds (default 40)
<code>-d, -debug</code>	<i>debuglevel</i>	0 = none, 1 = some, 2 = more, 3 = a lot
<code>-v, -version</code>		print version information
<code>-h, -help</code>		print this help

Arguments:

URL	URL of object to be deleted
-----	-----------------------------

This command deletes specified file, or all copies of it in case of meta-URL. For supported URLs please read "Protocols, Uniform Resource Locators (URL) And Extensions Supported in ARC".

For setting defaults, see the section about `.ngrc` (section 2.5).

1.12 *ngls*

The `ngls` command can be used to retrieve information about stored objects.

Usage: `ngls [options] URL`

Options:

<code>-l, -long</code>		show details of file
<code>-L, -locations</code>		show locations (aka TURLs) of file
<code>-t, -timeout</code>	<i>time</i>	timeout for MDS queries in seconds (default 40)
<code>-d, -debug</code>	<i>debuglevel</i>	0 = none, 1 = some, 2 = more, 3 = a lot
<code>-v, -version</code>		print version information
<code>-h, -help</code>		print this help

Arguments:

URL	URL of object for which information to be
-----	---

¹ Called `ngrm` in releases 0.5.29 and above

Arguments:

retrieved

This command retrieves common information about the specified object (usually, a file): size, date of creation, checksum, etc.

For setting defaults, see the section about `.ngrc` (section 2.5).

1.13 *ngacl*

The `ngacl` command can be used to manipulate permissions of stored objects.

Usage: `ngacl [options] get|set URL`

Options:

<code>-t, -timeout</code>	<i>time</i>	timeout for MDS queries in seconds (default 40)
<code>-d, -debug</code>	<i>debuglevel</i>	0 = none, 1 = some, 2 = more, 3 = a lot
<code>-v, -version</code>		print version information
<code>-h, -help</code>		print this help

Arguments:

<code>get</code>	Get ACL (currently only GACL is supported) for the object
<code>set¹</code>	set ACL (currently only GACL is supported) for the object
<code>URL</code>	Object URL; currently only GridFTP and SSE services are supported

This command may retrieve and modify access control information associated with stored object if service supports GridSite GACL language for access control. ACL document is printed to standard output and acquired from standard input.

For setting defaults, see the section about `.ngrc` (section 2.5).

1.14 *ngrequest²*

The `ngrequest` command initiates direct transfer of data between 2 servers (known as third-party transfer).

Usage: `ngrequest [options] destination`

Options:

<code>-S, -source</code>		URL to transfer data from
<code>-t, -timeout</code>	<i>time</i>	timeout for MDS queries in seconds (default 40)
<code>-d, -debug</code>	<i>debuglevel</i>	0 = none, 1 = some, 2 = more, 3 = a lot
<code>-v, -version</code>		print version information
<code>-h, -help</code>		print this help

1 Called `put` in releases 0.5.29 and above

2 Called `ngtransfer` in releases 0.5.29 and above

Arguments:

`destination` URL to which object to be copied/replicated

This command transfers or initiate transfer of objects specified by multiple `-source` option to `destination`. Currently supported combinations are between GridFTP servers and from any supported source to SSE service.

For setting defaults, see the section about `.ngrc` (section 2.5).

2 User configuration files

There are a few files located in the user's home directory that are used for configuring the ARC user interface. These files all start with `.ng`.

2.1 `.ngalias`

In this file a user can specify a set of aliases that can be used instead of host names to identify clusters. An alias can represent a single cluster or a set of clusters. An alias definition can contain another alias defined earlier in the file. With this sample `.ngalias` file:

```
# Sample .ngalias file
# Comments starts with #
nglund=quark.hep.lu.se
ngoslo=Grid.uio.no
myng=nglund ngoslo
```

the option `-c myng` resolves to `-c quark.hep.lu.se -c Grid.uio.no`.

2.2 `.nggiislist`

The user can specify a personal `giislist` file different from the system wide default by creating the file `.nggiislist` in the home directory. See the discussion in the `ngsub` section (section 1.1) for the format of this file.

2.3 `.nghistory`

This file contains the `jobIDs` of the jobs the user has submitted together with the time of submission. This file is purely informational.

2.4 `.ngjobs`

This a local list of the user's active jobs. When a job is successfully submitted, it is added to this list, and when it is removed from the remote cluster, it is removed from this list. This list is used as the list of all active jobs when the user specifies `-all` option to the various ARC user interface commands. For information about how to reconstruct this file in case it is damaged or you relocate to a different workstation, see section 1.9 about the `ngsync` command.

2.5 `.ngrc`

This file contains user's default settings for the debug level, the information system query timeout and the download directory used by `ngget` (see section 1.4). A sample

file could be the following:

```
# Sample .nsrc file
# Comments starts with #
NGDEBUG=1
NGTIMEOUT=60
NGDOWNLOAD=/tmp
```

If the file is missing or some options are not specified in the file, the system defaults are used. The default debug level is 0, the default information system timeout is 40 seconds and the default download directory is the current working directory.

If the environment variables NGDEBUG, NGTIMEOUT or NGDOWNLOAD are defined, these take precedence over the values defined in `.nsrc`. Any command line options overrides the defaults.