|  |
| --- |
| NCBI |
| NetSchedule Protocol |
| Informal description |
|  |
| **Sergey Satskiy** |
| **1/10/2012** |

|  |
| --- |
| Document version: 1.70 |

Table of Contents

[NetSchedule Protocol 7](#_Toc464139623)

[Overview 7](#_Toc464139624)

[Authentication Line 8](#_Toc464139625)

[Queue Name Line 9](#_Toc464139626)

[Command Line 10](#_Toc464139627)

[NetSchedule Response Line or Lines 10](#_Toc464139628)

[NetSchedule Commands 10](#_Toc464139629)

[Maintenance Commands 11](#_Toc464139630)

[The SHUTDOWN Command 11](#_Toc464139631)

[The GETCONF Command 11](#_Toc464139632)

[The VERSION command 12](#_Toc464139633)

[The HEALTH command 13](#_Toc464139634)

[The ACKALERT command 14](#_Toc464139635)

[The SETCLIENTDATA command 15](#_Toc464139636)

[The SETSCOPE command 16](#_Toc464139637)

[The RECO Command 17](#_Toc464139638)

[The ACNT Command 18](#_Toc464139639)

[The QLST Command 19](#_Toc464139640)

[The QINF Command 19](#_Toc464139641)

[The QINF2 Command 20](#_Toc464139642)

[The DROPQ command 21](#_Toc464139643)

[The QCRE command 22](#_Toc464139644)

[The QDEL command 22](#_Toc464139645)

[The STATUS command 23](#_Toc464139646)

[The STATUS2 command 24](#_Toc464139647)

[The STAT command 26](#_Toc464139648)

[The STAT CLIENTS command 27](#_Toc464139649)

[The STAT NOTIFICATIONS command 29](#_Toc464139650)

[The STAT AFFINITIES command 30](#_Toc464139651)

[The STAT JOBS command 32](#_Toc464139652)

[The STAT GROUPS command 33](#_Toc464139653)

[The STAT SCOPES command 34](#_Toc464139654)

[The STAT QCLASSES command 35](#_Toc464139655)

[The STAT QUEUES command 36](#_Toc464139656)

[The STAT SERVICES command 37](#_Toc464139657)

[The STAT ALERTS command 37](#_Toc464139658)

[The DUMP command 38](#_Toc464139659)

[The AFLS command 40](#_Toc464139660)

[The GETP command 41](#_Toc464139661)

[The GETP2 command 42](#_Toc464139662)

[The GETC command 43](#_Toc464139663)

[The CANCELQ command 44](#_Toc464139664)

[The REFUSESUBMITS command 45](#_Toc464139665)

[The QPAUSE command 45](#_Toc464139666)

[The QRESUME command 46](#_Toc464139667)

[The SETQUEUE command 46](#_Toc464139668)

[Submitter / Worker Node / Reader Common Commands 48](#_Toc464139669)

[The QUIT command 48](#_Toc464139670)

[Submitter Commands 49](#_Toc464139671)

[The MGET command 49](#_Toc464139672)

[The SST command 49](#_Toc464139673)

[The SST2 command 50](#_Toc464139674)

[The SUBMIT command 52](#_Toc464139675)

[The LISTEN command 53](#_Toc464139676)

[The CANCEL command 55](#_Toc464139677)

[The BSUB command 56](#_Toc464139678)

[Worker Node Commands 59](#_Toc464139679)

[The MPUT command 59](#_Toc464139680)

[The CLRN command 59](#_Toc464139681)

[The WST command 60](#_Toc464139682)

[The WST2 command 61](#_Toc464139683)

[The CHAFF command 62](#_Toc464139684)

[The SETAFF command 63](#_Toc464139685)

[The GET command 64](#_Toc464139686)

[The GET2 command 66](#_Toc464139687)

[The PUT command 68](#_Toc464139688)

[The PUT2 command 69](#_Toc464139689)

[The RETURN command 70](#_Toc464139690)

[The RETURN2 command 71](#_Toc464139691)

[The RESCHEDULE command 72](#_Toc464139692)

[The REDO command 73](#_Toc464139693)

[The WGET command 73](#_Toc464139694)

[The CWGET command 75](#_Toc464139695)

[The FPUT command 75](#_Toc464139696)

[The FPUT2 command 76](#_Toc464139697)

[The JXCG command 77](#_Toc464139698)

[The JDEX command 79](#_Toc464139699)

[Reader Commands 80](#_Toc464139700)

[The READ command 80](#_Toc464139701)

[The READ2 command 82](#_Toc464139702)

[The CWREAD command 84](#_Toc464139703)

[The SETRAFF command 85](#_Toc464139704)

[The CHRAFF command 86](#_Toc464139705)

[The CFRM command 87](#_Toc464139706)

[The FRED command 87](#_Toc464139707)

[The RDRB command 88](#_Toc464139708)

[The REREAD command 89](#_Toc464139709)

[Obsolete Commands 91](#_Toc464139710)

[Notification Formats 91](#_Toc464139711)

[Job State Changed Notification Format 92](#_Toc464139712)

[Job Available Notification Format 93](#_Toc464139713)

History of Changes

|  |  |  |  |
| --- | --- | --- | --- |
| Document version | Date | Author | Description |
| 1.10 | Mar 5, 2012 | Sergey Satskiy | Jobs grouping support commands |
| 1.11 | Mar 6, 2012 | Sergey Satskiy | READ output, notification formats |
| 1.12 | Mar 15, 2012 | Sergey Satskiy | Declare QPRT obsolete for NetSchedule 4.10.0 |
| 1.13 | Mar 19, 2012 | Sergey Satskiy | Declare STSN obsolete for NetSchedule 4.10.0  Introduce the affinity parameter for STAT JOBS |
| 1.14 | Mar 29, 2012 | Sergey Satskiy | New commands: SST2/WST2/STATUS2 |
| 1.15 | Apr 18, 2012 | Sergey Satskiy | Mark JXCG2 as available only in 4.10.0. It is removed from 4.11.0. |
| 1.16 | Apr 23, 2012 | Sergey Satskiy | Update the SST2/WST2/STATUS2 for NS 4.11.0.  Update the GET2 output for NS 4.11.0 |
| 1.17 | Apr 27, 2012 | Sergey Satskiy | New parameter for GET2 command for NS 4.11.0 |
| 1.18 | Apr 30, 2012 | Sergey Satskiy | Limit for err\_msg length for FPUT/FPUT2 commands for NS 4.11.0. |
| 1.19 | Apr 30, 2012 | Sergey Satskiy | Dima insisted on changing the position of the exclusive\_new\_aff option in the GET2 command. So it does. |
| 1.20 | May 2, 2012 | Sergey Satskiy | Adding REFUSESUBMITS command for NS 4.11.0 |
| 1.21 | May 2, 2012 | Sergey Satskiy | New fields in the STAT output |
| 1.22 | May 9, 2012 | Sergey Satskiy | Drained SHUTDOWN for NS 4.11.0  STAT JOBS for the server for NS 4.11.0 |
| 1.23 | May 14, 2012 | Sergey Satskiy | client\_host and client\_port handshake parameters description added |
| 1.24 | May 15, 2012 | Sergey Satskiy | The ‘client’ handshake parameter explanation update.  Updated description of the exclusive\_new\_aff parameter for GET2. |
| 1.25 | May 21, 2012 | Sergey Satskiy | Disconnect by NetSchedule initiative in case of command parsing errors. |
| 1.26 | June 1, 2012 | Sergey Satskiy | Added description of the SETQUEUE command for NS 4.12.0 |
| 1.27 | July 30, 2012 | Sergey Satskiy | New LISTEN command for NS 4.14.0  Fixed job state changed notification format. |
| 1.28 | August 24, 2012 | Sergey Satskiy | New QINF2 command for NS 4.14.0  Updated description of the RECO command. |
| 1.29 | August 28, 2012 | Sergey Satskiy | New ‘ip’ and ‘sid’ parameters for all the commands for NS 4.14.0 |
| 1.30 | November 13, 2012 | Sergey Satskiy | Bug fix: copy-paste bugs in SST2/WST2 description. |
| 1.31 | December 19, 2012 | Sergey Satskiy | New SETAFF command for NS 4.16.4 |
| 1.32 | February 1, 2013 | Sergey Satskiy | DROJ, FRES, QERY, QSEL, MONI, LOG have been deleted from NS 4.16.5 |
| 1.33 | June 5, 2013 | Sergey Satskiy | GETP2 added for NS 4.16.9 |
| 1.34 | June 13, 2013 | Sergey Satskiy | Removed description of commands which are not available in NS 4.16.8 and up |
| 1.35 | June 27, 2013 | Sergey Satskiy | Adding HEALTH command description for NS 4.16.10 and up. |
| 1.36 | December 31, 2013 | Sergey Satskiy | HEALTH command additions for NS 4.17.0  New ACKALERT command for NS 4.17.0  Service parameter for QINF2 command for NS 4.17.0 |
| 1.37 | March 14, 2014 | Sergey Satskiy | STAT SERVICES for NS 4.17.0 |
| 1.38 | March 19, 2014 | Sergey Satskiy | STAT ALERTS for NS 4.17.0 |
| 1.39 | March 20, 2014 | Sergey Satskiy | QPAUSE, QRESUME, RETURN2 and other changes related to queue pausing for NS 4.17.0 |
| 1.40 | March 26, 2014 | Sergey Satskiy | Adding ACKALERT parameter ‘user’ for NS 4.17.0 |
| 1.41 | March 31, 2014 | Sergey Satskiy | Adding SETCLIENTDATA command for NS 4.17.0 |
| 1.42 | April 1, 2014 | Sergey Satskiy | Changing output format of the STAT CLIENTS command |
| 1.43 | April 9, 2014 | Sergey Satskiy | Introducing ncbi\_phid parameter to all the commands of NS 4.17.0 |
| 1.44 | April 25, 2014 | Sergey Satskiy | Adding the ‘group’ parameter for the GET2 command for NS 4.17.1 |
| 1.45 | May 12, 2014 | Sergey Satskiy | Adding the ‘aff’ parameter for the CANCEL command fo NS 4.17.2 |
| 1.46 | May 21, 2014 | Sergey Satskiy | Bug fix: improper description of the QPAUSE command: the pullback/nopullback were exchanged. |
| 1.47 | June 3, 2014 | Sergey Satskiy | READ notifications for NS 4.17.2, fix the READ description |
| 1.48 | June 12, 2014 | Sergey Satskiy | RESCHEDULE for NS 4.19.0 |
| 1.49 | July 23, 2014 | Sergey Satskiy | Changes in the CANCEL command for NS 4.20.0; extending the DUMP command status parameter for NS 4.20.0; adding the aff parameter to the DUMP command for NS 4.20.0 |
| 1.50 | August 6, 2014 | Sergey Satskiy | Adding ‘no\_retries’ parameter of the FPUT2 and FRED commands for NS 4.20.0 |
| 1.51 | August 11, 2014 | Sergey Satskiy | Extended output for the CANCEL and CANCELQ commands for NS 4.20.0 |
| 1.52 | September 3, 2014 | Sergey Satskiy | Adding CWREAD, SETRAFF, CHRAFF for NS 4.20.0 |
| 1.53 | September 10, 2014 | Sergey Satskiy | READ2 for NS 4.20.0 |
| 1.54 | October 23, 2014 | Sergey Satskiy | Adding ClientType parameter description for the handshake phase for NS 4.20.1 |
| 1.55 | May 5, 2015 | Sergey Satskiy | Fixing error in the description of the no\_more\_jobs output parameter of the READ2 command. |
| 1.56 | May 20, 2015 | Sergey Satskiy | Changes in READ/READ2 for NS 4.22.0: two new Boolean flags |
| 1.57 | May 27, 2015 | Sergey Satskiy | Updated LISTEN output |
| 1.58 | June 16, 2015 | Sergey Satskiy | New prioritized\_aff flag for GET2/READ2 for NS 4.22.0 |
| 1.59 | June 18, 2015 | Sergey Satskiy | New ‘blacklist’ option for RDRB for NS 4.22.0 |
| 1.60 | August 19, 2015 | Sergey Satskiy | Group lists for GET2/READ/READ2 for NS 4.23.0 |
| 1.61 | October 26, 2015 | Sergey Satskiy | Adding a note about the command synopsis. |
| 1.62 | March 11, 2016 | Sergey Satskiy | Adding information about the ‘scope’ feature for NS 4.25.0 |
| 1.63 | September 16, 2016 | Sergey Satskiy | GET2/READ2 allowed any\_aff together with prioritized\_aff for NS 4.27.0 |
| 1.64 | September 27, 2016 | Sergey Satskiy | Fixed RECO output description |
| 1.65 | October 13, 2016 | Sergey Satskiy | Added REDO and REREAD commands for NS 4.28.0 |
| 1.66 | December 13, 2016 | Sergey Satskiy | Added ‘-‘ affinity and group for NS 4.29.0 |
| 1.67 | May 11, 2017 | Sergey Satskiy | Added GET2 output parameters (submitter\_notif\_host and submitter\_notif\_port) for NS 4.30.0 |
| 1.68 | November 17, 2017 | Sergey Satskiy | Addig need\_progress\_msg parameter for SST2/WST2/LISTEN/SUBMIT commands; adding need\_stolen for LISTEN command.  Job change notification format changes are described.  The changes are for NS-4.31.0. |
| 1.69 | November 20, 2017 | Sergey Satskiy | Adding need\_progress\_msg parameter for STATUS2 command for NS 4.31.0 |
| 1.70 | November 23, 2018 | Sergey Satskiy | Adding new DUMP options for NS 4.42.0 |

# NetSchedule Protocol

This document describes the NetSchedule server protocol in detail.

## Overview

The NetSchedule protocol is based on human readable strings exchanged over a TCP/IP connection. An exchange unit is a line which is ended by a carriage return character. The figure below shows the overall structure of the protocol.



Protocol Structure

A session of communication with a NetSchedule server starts with opening a TCP/IP connection.

The first phase – handshaking – is comprised of two lines which a NetSchedule server expects from a client. The first line is what a client tells NetSchedule about itself. The second line is a queue name and this is the queue to which all the other commands in the session will be applied to. Neither of these two lines causes a NetSchedule server to generate any output (except for errors, if there are any).

The second phase is an arbitrary number of command lines followed by NetSchedule server output. Some commands cause a single line output while the others may cause a multiline output.

A command may require certain privileges and/or an existing queue name. These requirements will be indicated in the description for the command.

A session can3 be finished by an explicit command (QUIT) or by closing the connection.

NetSchedule 4.11.0 and up will close a connection by its own initiative when it detects command parsing errors.

**Note:** An important detail is that NetSchedule will silently ignore unrecognized argument.

### Authentication Line

The idea of this line is that a client can tell a NetSchedule server what the client is. Here is the synopsis of the authentication line:













|  |  |
| --- | --- |
| Parameter | Description |
| ClientID | Client name.  This name is used to detect if the client has administrative privileges.  NetSchedule below 4.9.0 uses hardcoded client names ‘netschedule\_control’ and ‘netschedule\_admin’ to provide the admin privileges.  NetSchedule 4.9.0 and up use a configuration parameter to get a list of the client names to grant the admin privileges. |
| ProgramName | Program name.  Usually it is the name of the binary which connects to the server. No decisions are made based on the value of this parameter. |
| ClientNodeID | Identifier of the client node.  NetSchedule server has a client registry and the value of this parameter is used as a key in the registry. It is highly recommended to provide this identifier. Otherwise some services will not be provided for the client.  If client\_node is provided then client\_session must also been provided. |
| ClientSessionID | Identifier of the client session.  The same identifier should be used throughout the lifetime of the node, but should be changed if the node has restarted. The NetSchedule server associates a client with a session and some other data, such as a list of executing jobs. If the client session is changed then NetSchedule will reset the data for the client - for example, by moving the jobs formerly in the executing jobs list to the pending state. |
| TypeID | Allowed values (case insensitive): admin, submitter, worker node, reader, auto, reset.  By default NetSchedule server detects and displays the client type in accordance with what commands were issued by the client. It can be overridden by the ‘admin’, ‘submitter’, ‘worker node’ and ‘reader’ client types. If given then the client type is displayed accordingly. The value ‘auto’ instructs NetSchedule to fallback to displaying the type basing on commands. The value ‘reset’ instructs NetSchedule to reset the collected client type information (historical data) and switch to auto detection.  **Note**: introduced in NS 4.17.2 (admin) and in NS 4.20.1 (all others) |
| ScopeID | Identifier of the scope (optional).  Three cases are recognized:   * Empty string * Fixed string no-scope-only * All other string identifiers   The scope affects how the commands related to jobs are executed. The provided scope limits the considered set of jobs appropriately. The scope could also be changed at any time using the SETSCOPE command.  **Note**: introduced in NS-4.25.0 |

Two more parameters ‘client\_host’ and ‘control\_port’ (optional, available for NS 4.11.0 and up) are processes in a special way. They are stored in the client registry and can be seen later using the STAT CLIENTS command.

### Queue Name Line

The idea of this line is to specify what jobs queue the client is interested in. Here is the synopsis of the queue line:



In some cases a queue name is not required and for these cases a queue name can be empty or the special hardcoded value ‘noname’.

### Command Line

A command line is always a single line. The format will be described separately for each command.

### NetSchedule Response Line or Lines

The NetSchedule response could be a single line or a multiline response.

The single line response synopsis is as follows:

OK:[WARNING:Warning1Description;[Warning2Description[…]]]

The multiline NetSchedule response is as follows:

[OK:][information] \*

OK:END

There can be many empty or non-empty lines with or without the “OK:” prefix. The last line of a multiline response is always “OK:END”.

In case of errors for single or multiline responses, the NetSchedule server generates the output as follows:

ERR:[ErrorDescription]

The error describing line is the last one if an error appeared.

# NetSchedule Commands

This section describes each of the supported commands in detail.

The commands are grouped into sections as they are usually used. This grouping does not mean that a certain type of a client is restricted to execute a command from another type of client group. The grouping is provided for convenience only, not based on access rights permissions.

**Note**: the sequence of the arguments in the synopsis of each command matters. The argument description sequence in tables below does not matter.

## Maintenance Commands

### The SHUTDOWN Command

**Privileges**: admin privileges are required.

**Requires a queue**: no

**NetSchedule output type**: single line

**Version**: See the parameters description

**Synopsis**:

SHUTDOWN [Drain] [IP] [SID] [PHID]

**Description**: initiates the server shutdown. This may take time depending on how much data need to be flushed from a memory cache to a database.

|  |  |  |
| --- | --- | --- |
| Parameter | Name | Description |
| Drain | drain | Integer value: 0 or 1.  Default: 0  If 1 is given then the server will switch to the drained shutdown mode. That means that all the further submits will be refused and the server will wait till the last job is wiped out from the NetSchedule data structures. As soon as there are no jobs the server will be shut down.  **Note**: Introduced in version 4.11.0. |
| IP | ip | IP address of a remote client  **Note**: Introduced in version 4.14.0. |
| SID | sid | Session ID of a remote client  **Note**: Introduced in version 4.14.0. |
| PHID | ncbi\_phid | CGI page hit ID  **Note**: Introduced in version 4.17.0 |

### The GETCONF Command

**Privileges**: admin privileges are required.

**Requires a queue**: no

**NetSchedule output type**: multiple lines

**Version**: requires NetSchedule version 4.10.0 and up

**Synopsis**:

GETCONF [IP] [SID] [PHID]

**Description**: Prints the server configuration. Please note that it prints the current content of the configuration file, not the current effective values of the parameters. For example, the configuration file is read when the server is launched, but later the configuration file could be altered. Then the RECO command could be issued, causing the configuration file to be re-read. However, the server might not be able to update some parameters, so they will retain their original values. The GETCONF command will print the latest content of the configuration file, which in this sort of scenario doesn't match the current effective values of the parameters.

|  |  |  |
| --- | --- | --- |
| Parameter | Name | Description |
| IP | ip | IP address of a remote client  **Note**: Introduced in version 4.14.0. |
| SID | sid | Session ID of a remote client  **Note**: Introduced in version 4.14.0. |
| PHID | ncbi\_phid | CGI page hit ID  **Note**: Introduced in version 4.17.0 |

**Output synopsis**:

OK:<configuration file content>

### The VERSION command

**Privileges**: any.

**Requires a queue**: no

**NetSchedule output type**: single line

**Synopsis**:

VERSION [IP] [SID] [PHID]

**Description**: provides the version of different NetSchedule components along with the server node and session identifiers. The output format differs between versions: versions below 4.10.0 used a space separated format while versions beginning with 4.10.0 use a URL encoded format.

|  |  |  |
| --- | --- | --- |
| Parameter | Name | Description |
| IP | ip | IP address of a remote client  **Note**: Introduced in version 4.14.0. |
| SID | sid | Session ID of a remote client  **Note**: Introduced in version 4.14.0. |
| PHID | ncbi\_phid | CGI page hit ID  **Note**: Introduced in version 4.17.0 |

**Output synopsis**:

OK:<URL encoded key=value pairs>

**Example**:

VERSION

OK:server\_version=0.0.0&storage\_version=4.3.1&protocol\_version=1.4.0&build\_date=Jan+10+2012+14%3A04%3A48&ns\_node=dev\_4\_10\_0&ns\_session=16871957761332

### The HEALTH command

**Privileges**: any.

**Requires a queue**: no

**NetSchedule output type**: single line

**Version**: 4.16.10 and up

**Synopsis**:

HEALTH [IP] [SID] [PHID]

**Description**: provides the process health associated information which includes but not limited to memory, CPU, threads and file descriptors information. The other information fields may be added to the output later.

|  |  |  |
| --- | --- | --- |
| Parameter | Name | Description |
| IP | ip | IP address of a remote client |
| SID | sid | Session ID of a remote client |
| PHID | ncbi\_phid | CGI page hit ID  **Note**: Introduced in version 4.17.0 |

**Output synopsis**:

OK:<URL encoded key=value pairs>

**Example**:

HEALTH

OK:pid=32384&ns\_node=dev\_4\_10\_0&ns\_session=16871957761332&cpu\_count=24&started=06%2F28%2F2013+10%3A01%3A19&user\_time=0&system\_time=0.01&physical\_memory=50740670464&mem\_used\_total=863481856&mem\_used\_resident=60162048&mem\_used\_shared=2789376&proc\_fd\_soft\_limit=8192&proc\_fd\_hard\_limit=8192&proc\_fd\_used=45&proc\_thread\_count=11

**Note**: NS version 4.17.0 and up has additional alert output values, e.g.

alert\_<alert id>=<integer value>

where alert id is a string identifier. Currently there is one supported identifier ‘config’ though the future NetSchedule versions may extend this list.

The absolute alert value is the number of times the alert has been detected. If the value is positive then the alert has not been acknowledged yet. If the alert has been acknowledged then the value is negative.

### The ACKALERT command

**Privileges**: any.

**Requires a queue**: no

**NetSchedule output type**: single line

**Version**: 4.17.0 and up

**Synopsis**:

ACKALERT [ALERT] [USER] [IP] [SID] [PHID]

**Description**: acknowledges the required alert.

|  |  |  |
| --- | --- | --- |
| Parameter | Name | Description |
| ALERT | alert | Alert identifier to be acknowledged. The alert identifier is not case sensitive. |
| USER | user | User identifier who acknowledges the alert. |
| IP | ip | IP address of a remote client |
| SID | sid | Session ID of a remote client |
| PHID | ncbi\_phid | CGI page hit ID  **Note**: Introduced in version 4.17.0 |

**Output synopsis**:

OK:

**Example**:

ACKALERT config kazimird

OK:

**Note**: The output may have warnings. The warnings appear in cases if the alert is not found or if the alert has already been acknowledged.

### The SETCLIENTDATA command

**Privileges**: any.

**Requires a queue**: yes

**NetSchedule output type**: single line

**Version**: 4.17.0 and up

**Synopsis**:

SETCLIENTDATA DATA [VERSION] [IP] [SID] [PHID]

**Description**: sets the non-anonymous client data. The data are transient and will not survive restart.

|  |  |  |
| --- | --- | --- |
| Parameter | Name | Description |
| DATA | data | Arbitrary data to be saved for the client. |
| VERSION | version | Integer version of the data to update. If -1 (default) then the data are updated unconditionally. Otherwise the current version is compared with the given. If they mismatch the client data are not updated and an error is generated.  The data and its version could be seen in the STAT CLIENTS output. |
| IP | ip | IP address of a remote client |
| SID | sid | Session ID of a remote client |
| PHID | ncbi\_phid | CGI page hit ID  **Note**: Introduced in version 4.17.0 |

**Output synopsis**:

OK:version=<VER>

|  |  |
| --- | --- |
| Parameter | Description |
| VER | Version of the written data client. |

**Example**:

SETCLIENTDATA data=”My data”

OK:version=1

**Note**: The output may have errors. The errors appear in cases if the data are larger than a configured limit, when an anonymous client is tried to set data, when there is no queue or when the client data version mismatch the provided one.

### The SETSCOPE command

**Privileges**: any.

**Requires a queue**: yes

**NetSchedule output type**: single line

**Version**: 4.25.0 and up

**Synopsis**:

SETSCOPE [SCOPE] [IP] [SID] [PHID]

**Description**: sets the scope for the further command execution. The provided scope is effective till the connection is closed or till another SETSCOPE is received. If scope identifier is not provided then the scope is reset to an empty string.

|  |  |  |
| --- | --- | --- |
| Parameter | Name | Description |
| SCOPE | scope | Scope identifier for the further commands. Three cases are recognized:   * Empty string * Fixed string no-scope-only * Any other string identifier |
| IP | ip | IP address of a remote client |
| SID | sid | Session ID of a remote client |
| PHID | ncbi\_phid | CGI page hit ID |

**Output synopsis**:

OK:

**Example**:

SETSCOPE scope=MyScope

OK:

**Note**: The output may have errors. The errors appear in cases if there are no vacant slots in the queue scope registry.

### The RECO Command

**Privileges**: any – prior NS 4.10.0, admin – NS 4.10.0 and up

**Requires a queue**: no

**NetSchedule output type**: single line

**Synopsis**:

RECO [IP] [SID] [PHID]

**Description**: Reads the configuration file if it was changed. Accepts all the new parameters which can be changed without restarting the server.

|  |  |  |
| --- | --- | --- |
| Parameter | Name | Description |
| IP | ip | IP address of a remote client  **Note**: Introduced in version 4.14.0. |
| SID | sid | Session ID of a remote client  **Note**: Introduced in version 4.14.0. |
| PHID | ncbi\_phid | CGI page hit ID  **Note**: Introduced in version 4.17.0 |

**Output synopsis (NS prior 4.14.0)**:

OK:

**Output synopsis (NS 4.14.0 – NS 4.21.0)**:

OK: WARNING:No changeable parameters were identified in the new cofiguration file;

or

OK:WARNING:Configuration file has not been changed, RECO ignored;

or

OK:<exact description of what was taken into account>

e.g.:

OK:"deleted\_queue\_classes" ["class1"], "queue\_class\_changes" {"class2" {"timeout" [1, 2], "description" ["class two", "class two updated"]}}, "deleted\_queues" ["q1"], "queue\_changes" {"q2" {"max\_input\_size" [33, 77], "description" ["class two", "class two updated"]}}

**Output synopsis (NS 4.21.1 and up):**

OK:WARNING:eNoParametersChanged:No changes in changeable parameters were identified in the new cofiguration file;

or

OK:WARNING:eConfigFileNotChanged:Configuration file has not been changed, RECO ignored;

or

OK:<serialized JSON with a description of what was changed>

e.g.:

{"added\_queues": {"raid\_remote": "default"}}

### The ACNT Command

**Privileges**: any.

**Requires a queue**: no

**NetSchedule output type**: single line

**Synopsis**:

ACNT [IP] [SID] [PHID]

**Description**: Provides the number of the jobs in the pending and running states summarized for all the current queues.

|  |  |  |
| --- | --- | --- |
| Parameter | Name | Description |
| IP | ip | IP address of a remote client  **Note**: Introduced in version 4.14.0. |
| SID | sid | Session ID of a remote client  **Note**: Introduced in version 4.14.0. |
| PHID | ncbi\_phid | CGI page hit ID  **Note**: Introduced in version 4.17.0 |

**Output synopsis**:

OK:<Count>

|  |  |
| --- | --- |
| Parameter | Description |
| Count | Integer: the number of jobs in the pending and running states summarized for all the current queues |

**Example**:

ACNT

OK:674

### The QLST Command

**Privileges**: any.

**Requires a queue**: no

**NetSchedule output type**: single line

**Synopsis**:

QLST [IP] [SID] [PHID]

**Description**: Provides the list of the existed queues.

|  |  |  |
| --- | --- | --- |
| Parameter | Name | Description |
| IP | ip | IP address of a remote client  **Note**: Introduced in version 4.14.0. |
| SID | sid | Session ID of a remote client  **Note**: Introduced in version 4.14.0. |
| PHID | ncbi\_phid | CGI page hit ID  **Note**: Introduced in version 4.17.0 |

**Output synopsis**:

OK:<Queues>

|  |  |
| --- | --- |
| Parameter | Description |
| Queues | Semicolon separated list of the NetSchedule queues |

**Example**:

QLST

OK:TEST2;TEST3;

### The QINF Command

**Privileges**: any.

**Requires a queue**: no

**NetSchedule output type**: single line

**Synopsis**:

QINF <QueueName> [IP] [SID] [PHID]

**Description**: Provides the tab separated queue info.

|  |  |  |
| --- | --- | --- |
| Parameter | Name | Description |
| QueueName | qname | Queue name |
| IP | ip | IP address of a remote client  **Note**: Introduced in version 4.14.0. |
| SID | sid | Session ID of a remote client  **Note**: Introduced in version 4.14.0. |
| PHID | ncbi\_phid | CGI page hit ID  **Note**: Introduced in version 4.17.0 |

**Output synopsis**:

OK:<QueueKind> <QueueClass> <”QueueComment”>

|  |  |
| --- | --- |
| Parameter | Description |
| QueueKind | Integer value:  0 – static queue  1 – dynamic queue |
| QueueClass | String identifier: queue class |
| QueueComment | String: queue comment |

**Example**:

QINF TEST2

OK:0 TEST2 "TEST2 queue comment"

### The QINF2 Command

**Privileges**: any.

**Requires a queue**: no

**NetSchedule output type**: single line

**Version**: 4.14.0 and up

**Synopsis**:

**NS 4.14.0 and up**: QINF2 <QueueName> [IP] [SID]

**NS 4.17.0 and up**: QINF2 <QueueName | Service> [IP] [SID] [PHID]

**Description**: Provides all the queue parameters in a URL encoded format.

|  |  |  |
| --- | --- | --- |
| Parameter | Name | Description |
| QueueName | qname | Queue name |
| Service | service | Service name to be resolved to a queue (see ‘service\_to\_queue’ section in .ini file)  **Note**: Introduced in version 4.17.0. |
| IP | ip | IP address of a remote client  **Note**: Introduced in version 4.14.0. |
| SID | sid | Session ID of a remote client  **Note**: Introduced in version 4.14.0. |
| PHID | ncbi\_phid | CGI page hit ID  **Note**: Introduced in version 4.17.0 |

**Output synopsis**:

OK:<URL encoded queue parameters>

**Example**:

QINF2 MYQUEUE

OK:kind=static&position=0&qclass=&delete\_request=false&timeout=4000&notif\_hifreq\_interval=0.1&notif\_hifreq\_period=5&notif\_lofreq\_mult=50&dump\_buffer\_size=100&run\_timeout=600&program\_name=&failed\_retries=50&blacklist\_time=0&max\_input\_size=1000000&max\_output\_size=1000000&subm\_hosts=&wnode\_hosts=&wnode\_timeout=40&pending\_timeout=604800&max\_pending\_wait\_timeout=0&description=&run\_timeout\_precision=50&refuse\_submits=false

**Note:**

NS 4.17.0 and up adds ‘pause’ output parameter with values ‘nopause’, ‘nopullback’ or ‘pullback’.

### The DROPQ command

**Privileges**: queue administrator.

**Requires a queue**: yes

**NetSchedule output type**: single line

**Synopsis**:

DROPQ [IP] [SID] [PHID]

**Description**: deletes all the jobs from the queue.

|  |  |  |
| --- | --- | --- |
| Parameter | Name | Description |
| IP | ip | IP address of a remote client  **Note**: Introduced in version 4.14.0. |
| SID | sid | Session ID of a remote client  **Note**: Introduced in version 4.14.0. |
| PHID | ncbi\_phid | CGI page hit ID  **Note**: Introduced in version 4.17.0 |

**Output synopsis**:

OK:

### The QCRE command

**Privileges**: dynamic queue class administrator.

**Requires a queue**: no

**NetSchedule output type**: single line

**Synopsis**:

QCRE <QueueName> <QueueClass> [QueueComment] [IP] [SID] [PHID]

**Description**: creates a new dynamic queue.

|  |  |  |
| --- | --- | --- |
| Parameter | Name | Description |
| QueueName | qname | String identifier: the new queue name |
| QueueClass | qclass | String identifier: queue class |
| QueueComment | comment | String: queue comment |
| IP | ip | IP address of a remote client  **Note**: Introduced in version 4.14.0. |
| SID | sid | Session ID of a remote client  **Note**: Introduced in version 4.14.0. |
| PHID | ncbi\_phid | CGI page hit ID  **Note**: Introduced in version 4.17.0 |

**Output synopsis**:

OK:

### The QDEL command

**Privileges**: dynamic queue class administrator.

**Requires a queue**: no

**NetSchedule output type**: single line

**Synopsis**:

QDEL <QueueName> [IP] [SID] [PHID]

**Description**: deletes the given dynamic queue.

|  |  |  |
| --- | --- | --- |
| Parameter | Name | Description |
| QueueName | qname | String identifier: the name of a dynamic queue to be deleted. |
| IP | ip | IP address of a remote client  **Note**: Introduced in version 4.14.0. |
| SID | sid | Session ID of a remote client  **Note**: Introduced in version 4.14.0. |
| PHID | ncbi\_phid | CGI page hit ID  **Note**: Introduced in version 4.17.0 |

**Output synopsis**:

OK:

### The STATUS command

**Privileges**: any.

**Requires a queue**: yes

**NetSchedule output type**: single line

**Synopsis**:

STATUS <JobKey> [IP] [SID] [PHID]

**Description**: provides some information about the given job. The command is obsolete starting from NetSchedule 4.10.0, use STATUS2 instead.

|  |  |  |
| --- | --- | --- |
| Parameter | Name | Description |
| JobKey | job\_key | String identifier: the job key. |
| IP | ip | IP address of a remote client  **Note**: Introduced in version 4.14.0. |
| SID | sid | Session ID of a remote client  **Note**: Introduced in version 4.14.0. |
| PHID | ncbi\_phid | CGI page hit ID  **Note**: Introduced in version 4.17.0 |

**Output synopsis**:

OK:<Status> <RetCode> <”Output”> <”ErrorMessage”> <”Input”>

|  |  |
| --- | --- |
| Parameter | Description |
| Status | Integer value: job state  0 – pending  1 – running  3 – canceled  4 – failed  5 – done  6 – reading  7 – confirmed  8 – read failed |
| RetCode | Integer: return code from the last job event. |
| Output | Escaped output |
| ErrorMessage | Escaped error message from the last job event |
| Input | Escaped input |

Please note that a job may have many events during its lifetime and each event has a return code and error message fields (among the other fields). The STATUS command picks the return code and error message fields from the last event record.

**Example**:

STATUS JSID\_01\_1\_130.14.24.194\_9102

OK:0 0 "" "" "myinput"

### The STATUS2 command

**Privileges**: any.

**Requires a queue**: yes

**NetSchedule output type**: single line

**Version**: 4.10.0 and up, see also description of the output parameters

**Synopsis**:

STATUS2 <JobKey> [IP] [SID] [PHID] [NeedProgressMessage]

**Description**: provides some information about the given job.

|  |  |  |
| --- | --- | --- |
| Parameter | Name | Description |
| JobKey | job\_key | String identifier: the job key. |
| IP | ip | IP address of a remote client  **Note**: Introduced in version 4.14.0. |
| SID | sid | Session ID of a remote client  **Note**: Introduced in version 4.14.0. |
| PHID | ncbi\_phid | CGI page hit ID  **Note**: Introduced in version 4.17.0 |
| NeedProgressMessage | need\_progress\_msg | Tells if the command output should also have the job progress message. The accepted values are 0 and 1. Default is 0. If set to 1 then the output will have the ‘msg’ parameter, see below.  **Note:** Introduces in version 4.31.0 |

**Output synopsis**:

OK:job\_status=<Status>&job\_exptime=<AbsTime>&ret\_code=<RetCode>&output=<Output>&err\_msg=<ErrorMessage>&input=<Input>[&pause=<PauseStatus>[&msg=<ProgressMessage]]

|  |  |
| --- | --- |
| Parameter | Description |
| Status | The job state as string. It could be one of the following:  Pending  Running  Canceled  Failed  Done  Reading  Confirmed  ReadFailed |
| RetCode | Integer: return code from the last job event. |
| Output | URL encoded output |
| ErrorMessage | URL encoded error message from the last job event |
| Input | URL encoded input |
| AbsTime | **Note:** available for NS 4.11.0 and up  Time in seconds when a job is deleted if nothing else happens to the job. The time is an absolute UNIX time.  For the jobs in the Running and Pending states this value is calculated as the current time plus the queue timeout. |
| PauseStatus | **Note:** available for NS 4.17.0 and up  The parameter appears only of the corresponding queue is paused some way. The values are:  pullback  nopullback |
| ProgressMessage | **Note:** available for NS 4.31.0 and up.  The parameter appears only if the command has NeedProgressMessage parameter set to 1.  The value of the “ProgressMessage” is a URL encoded string. |

Please note that a job may have many events during its lifetime and each event has a return code and error message fields (among the other fields). The STATUS2 command picks the return code and error message fields from the last event record.

**Note**: if there is no such a job, NS 4.11.0 output will be as follows:

ERR:eJobNotFound:

**Example**:

STATUS2 JSID\_01\_1\_130.14.24.194\_9102

OK:job\_status=Pending&ret\_code=0&output=&err\_msg=&input=myinput

### The STAT command

**Privileges**: any.

**Requires a queue**: yes/no

**NetSchedule output type**: multiple lines

**Synopsis**:

STAT [ALL] [IP] [SID] [PHID]

**Description**: provides the server status and statistics information. If a queue was not provided at the handshake phase (NetSchedule 4.10.0 and up) then the transition counters are printed for all the queues along with some other statistics counters. If a queue was provided at the handshaking stage then similar information is provided for a single queue (the one provided at the handshaking phase).

Starting from NetSchedule 4.11.0 STAT with no queue provided also prints the SubmitsDisabledEffective value which reflects the server level value of the refuse submits mode.

Starting from NetSchedule 4.11.0 STAT with a queue provided also prints the SubmitsDisabledPrivate value which reflects the queue level value of the refuse submits mode and the SubmitsDisabledEffective value which is calculated as logical OR of the queue and the server refuse submits modes.

|  |  |  |
| --- | --- | --- |
| Parameter | Name | Description |
| ALL |  | Fixed keyword.  If given then extended information is provided. The parameter is applicable only for the case when a queue is provided at the handshaking phase. |
| IP | ip | IP address of a remote client  **Note**: Introduced in version 4.14.0. |
| SID | sid | Session ID of a remote client  **Note**: Introduced in version 4.14.0. |
| PHID | ncbi\_phid | CGI page hit ID  **Note**: Introduced in version 4.17.0 |

**Example**: the queue has been provided

STAT

OK:Started: 01/10/2012 17:44:47

OK:Pending: 1

OK:Running: 0

OK:Canceled: 0

OK:Failed: 0

OK:Done: 0

OK:Reading: 0

OK:Confirmed: 0

OK:ReadFailed: 0

OK:[Configured job submitters]:

OK:[Configured workers]:

OK:[Transitions counters]:

OK:submits: 1

OK:dbdeletions: 0

OK:Pending\_Running: 0

OK:Pending\_Canceled: 0

OK:Pending\_Done: 0

. . .

OK:garbage\_jobs: 0

OK:END

**Example**: the queue has not been provided

STAT

OK:Started: 01/10/2012 17:44:47

OK:[queue TEST2]

OK:submits: 1

OK:dbdeletions: 0

OK:Pending\_Running: 0

. . .

OK:garbage\_jobs: 0

OK:[queue TEST3]

OK:submits: 0

OK:dbdeletions: 0

OK:Pending\_Running: 0

OK:Pending\_Canceled: 0

. . .

OK:Reading\_ReadFailed\_timeout: 0

OK:garbage\_jobs: 0

OK:END

### The STAT CLIENTS command

**Privileges**: any.

**Requires a queue**: yes

**NetSchedule output type**: multiple lines

**Version**: 4.10.0 and up

**Synopsis**:

STAT CLIENTS [VERBOSE] [IP] [SID] [PHID]

**Description**: prints the clients registry.

|  |  |  |
| --- | --- | --- |
| Parameter | Name | Description |
| VERBOSE |  | Fixed keyword.  If given then more detailed information is provided about each client. Getting the detailed information is more expensive in term of the resources. |
| IP | ip | IP address of a remote client  **Note**: Introduced in version 4.14.0. |
| SID | sid | Session ID of a remote client  **Note**: Introduced in version 4.14.0. |
| PHID | ncbi\_phid | CGI page hit ID  **Note**: Introduced in version 4.17.0 |

**Example**: basic information

STAT CLIENTS

OK:CLIENT: mwebdev34:7600

OK: STATUS: active

OK: LAST ACCESS: 01/11/2012 10:36:08

OK: ADDRESS: localhost

OK: SESSION: 123456

OK: TYPE: worker node

OK: NUMBER OF SUBMITTED JOBS: 0

OK: RUNNING JOBS:

OK: JSID\_01\_1\_130.14.24.194\_9102

OK: NUMBER OF JOBS GIVEN FOR EXECUTION: 1

OK: NUMBER OF JOBS GIVEN FOR READING: 0

OK: NUMBER OF PREFERRED AFFINITIES: 1

OK: DATA: ‘abc cde’

OK: DATA VERSION: 1

OK:END

**Example**: extended information

STAT CLIENTS VERBOSE

OK:CLIENT mwebdev34:7600

OK: STATUS: active

OK: LAST ACCESS: 01/11/2012 10:36:56

OK: ADDRESS: localhost

OK: SESSION: 123456

OK: TYPE: worker node

OK: NUMBER OF SUBMITTED JOBS: 0

OK: RUNNING JOBS:

OK: JSID\_01\_1\_130.14.24.194\_9102

OK: NUMBER OF JOBS GIVEN FOR EXECUTION: 1

OK: NUMBER OF JOBS GIVEN FOR READING: 0

OK: PREFERRED AFFINITIES:

OK: 100

OK: DATA: ‘abc cde’

OK: DATA VERSION: 1

OK:END

**Note**: client data are provided as “printable string” (see NStr class description in the C++ Toolkit code) in single quotes.

### The STAT NOTIFICATIONS command

**Privileges**: any.

**Requires a queue**: yes

**NetSchedule output type**: multiple lines

**Version**: 4.10.0 and up

**Synopsis**:

STAT NOTIFICATIONS [VERBOSE] [IP] [SID] [PHID]

**Description**: prints the notifications registry.

|  |  |  |
| --- | --- | --- |
| Parameter | Name | Description |
| VERBOSE |  | Fixed keyword.  If given then more detailed information is provided about each notification. Getting the detailed information is more expensive in term of the resources. |
| IP | ip | IP address of a remote client  **Note**: Introduced in version 4.14.0. |
| SID | sid | Session ID of a remote client  **Note**: Introduced in version 4.14.0. |
| PHID | ncbi\_phid | CGI page hit ID  **Note**: Introduced in version 4.17.0 |

**Example**: basic information

STAT NOTIFICATIONS

OK:CLIENT: 'mwebdev34:7600'

OK: RECEPIENT ADDRESS: localhost:6302

OK: LIFE TIME: 01/11/2012 10:43:42

OK: ANY JOB: FALSE

OK: EXPLICIT AFFINITIES: n/a (available in VERBOSE mode)

OK: USE PREFERRED AFFINITIES: FALSE

OK: ACTIVE: FALSE

OK: HIGH FREQUENCY LIFE TIME: n/a

OK: SLOW RATE ACTIVE: FALSE

OK:END

**Example**: extended information

STAT NOTIFICATIONS VERBOSE

OK:CLIENT: 'mwebdev34:7600'

OK: RECEPIENT ADDRESS: localhost:6302

OK: LIFE TIME: 01/11/2012 10:43:42

OK: ANY JOB: FALSE

OK: EXPLICIT AFFINITIES:

OK: 'a33'

OK: USE PREFERRED AFFINITIES: FALSE

OK: ACTIVE: FALSE

OK: HIGH FREQUENCY LIFE TIME: n/a

OK: SLOW RATE ACTIVE: FALSE

OK:END

### The STAT AFFINITIES command

**Privileges**: any.

**Requires a queue**: yes

**NetSchedule output type**: multiple lines

**Version**: 4.10.0 and up

**Synopsis**:

STAT AFFINITIES [VERBOSE] [IP] [SID] [PHID]

**Description**: prints the affinities registry.

**Note**: starting from NS 4.25.0 the command is affected by the current scope (see SETSCOPE). An empty scope means that all the jobs will be included. The no-scope-only scope means that only the jobs which were submitted with an empty scope will be included. All the other scope identifiers mean that the only jobs from a particular scope will be included.

**Note**: starting from NS 4.29.0 there will be no jobs without affinities. If a client submits a job without an affinity then the server silently converts it to the ‘-‘ affinity.

|  |  |  |
| --- | --- | --- |
| Parameter | Name | Description |
| VERBOSE |  | Fixed keyword.  If given then more detailed information is provided about each affinity. Getting the detailed information is more expensive in term of the resources. |
| IP | ip | IP address of a remote client  **Note**: Introduced in version 4.14.0. |
| SID | sid | Session ID of a remote client  **Note**: Introduced in version 4.14.0. |
| PHID | ncbi\_phid | CGI page hit ID  **Note**: Introduced in version 4.17.0 |

**Example**: basic information

STAT AFFINITIES

OK:AFFINITY: '99'

OK: ID: 1

OK: NUMBER OF JOBS: 1

OK: NUMBER OF CLIENTS (PREFERRED): 0

OK: NUMBER OF CLIENTS (EXPLICIT WGET): 0

OK:AFFINITY: '100'

OK: ID: 2

OK: NUMBER OF JOBS: 0

OK: NUMBER OF CLIENTS (PREFERRED): 1

OK: NUMBER OF CLIENTS (EXPLICIT WGET): 0

OK:END

**Example**: extended information

STAT AFFINITIES VERBOSE

OK:NUMBER OF ENTRIES: 2

OK:AFFINITY: '99'

OK: ID: 1

OK: JOBS:

OK: JSID\_01\_1\_130.14.24.194\_9102

OK: CLIENTS (PREFERRED): NONE

OK: CLIENTS (EXPLICIT WGET): NONE

OK:AFFINITY: '100'

OK: ID: 2

OK: JOBS: NONE

OK: CLIENTS (PREFERRED):

OK: mwebdev34:7600

OK: CLIENTS (EXPLICIT WGET): NONE

OK:END

### The STAT JOBS command

**Privileges**: any.

**Requires a queue**: yes

**NetSchedule output type**: multiple lines

**Version**: 4.10.0 and up

**Synopsis**:

STAT JOBS [Affinity] [Group] [IP] [SID] [PHID]

|  |  |  |
| --- | --- | --- |
| Parameter | Name | Description |
| Affinity | Aff | String identifier: if given then the output will be limited by jobs which have the given affinity.  **Note:** starting from NS 4.29.0 the ‘-‘ affinity could be used here. The ‘-‘ affinity is a functional equivalent of a job without an affinity. |
| Group | Group | String identifier: if given then the output will cover only the jobs belonging to the given group.  **Note:** starting from NS 4.29.0 the ‘-‘ group could be used here. The ‘-‘ group is a functional equivalent of a job without a group. |
| IP | Ip | IP address of a remote client  **Note**: Introduced in version 4.14.0. |
| SID | sid | Session ID of a remote client  **Note**: Introduced in version 4.14.0. |
| PHID | ncbi\_phid | CGI page hit ID  **Note**: Introduced in version 4.17.0 |

**Description**: prints the number of jobs per status. NetSchedule 4.11.0 and up also supports STAT JOBS for the server (if the queue is not given at the handshake stage). STAT JOBS for the server does not support any parameters and prints information about each registered queue with a proper header.

**Note**: starting from NS 4.25.0 the command is affected by the current scope (see SETSCOPE). An empty scope means that all the jobs will be included. The no-scope-only scope means that only the jobs which were submitted with an empty scope will be included. All the other scope identifiers mean that the only jobs from a particular scope will be included.

**Example**:

STAT JOBS

OK:Pending: 0

OK:Running: 1

OK:Canceled: 0

OK:Failed: 0

OK:Done: 0

OK:Reading: 0

OK:Confirmed: 0

OK:ReadFailed: 0

OK:Total: 1

OK:END

### The STAT GROUPS command

**Privileges**: any.

**Requires a queue**: yes

**NetSchedule output type**: multiple lines

**Version**: 4.10.0 and up

**Synopsis**:

STAT GROUPS [VERBOSE] [IP] [SID] [PHID]

**Description**: prints the job groups registry.

**Note**: starting from NS 4.25.0 the command is affected by the current scope (see SETSCOPE). An empty scope means that all the jobs will be included. The no-scope-only scope means that only the jobs which were submitted with an empty scope will be included. All the other scope identifiers mean that the only jobs from a particular scope will be included.

**Note**: starting from NS 4.29.0 there will be jobs without a group. If a client submitted a job without a group then the server silently assigns the ‘-‘ group to the job.

|  |  |  |
| --- | --- | --- |
| Parameter | Name | Description |
| VERBOSE |  | Fixed keyword.  If given then more detailed information is provided about each group. Getting the detailed information is more expensive in term of the resources. |
| IP | ip | IP address of a remote client  **Note**: Introduced in version 4.14.0. |
| SID | sid | Session ID of a remote client  **Note**: Introduced in version 4.14.0. |
| PHID | ncbi\_phid | CGI page hit ID  **Note**: Introduced in version 4.17.0 |

**Example**: basic information

STAT GROUPS

OK:GROUP: '000'

OK: ID: 2

OK: NUMBER OF JOBS: 6

OK:END

**Example**: extended information

STAT GROUPS VERBOSE

OK:GROUP: '000'

OK: ID: 2

OK: JOBS:

OK: JSID\_01\_3\_130.14.24.194\_9102

OK: JSID\_01\_4\_130.14.24.194\_9102

OK: JSID\_01\_5\_130.14.24.194\_9102

OK: JSID\_01\_6\_130.14.24.194\_9102

OK: JSID\_01\_7\_130.14.24.194\_9102

OK: JSID\_01\_8\_130.14.24.194\_9102

OK:END

### The STAT SCOPES command

**Privileges**: any.

**Requires a queue**: yes

**NetSchedule output type**: multiple lines

**Version**: 4.25.0 and up

**Synopsis**:

STAT SCOPES [VERBOSE] [IP] [SID] [PHID]

**Description**: prints the scope registry.

|  |  |  |
| --- | --- | --- |
| Parameter | Name | Description |
| VERBOSE |  | Fixed keyword.  If given then more detailed information is provided about each scope. Getting the detailed information is more expensive in term of the resources. |
| IP | ip | IP address of a remote client |
| SID | sid | Session ID of a remote client |
| PHID | ncbi\_phid | CGI page hit ID |

### The STAT QCLASSES command

**Privileges**: any.

**Requires a queue**: yes

**NetSchedule output type**: multiple lines

**Version**: 4.14.0 and up

**Synopsis**:

STAT QCLASSES [IP] [SID] [PHID]

**Description**: prints the queue classes.

|  |  |  |
| --- | --- | --- |
| Parameter | Name | Description |
| IP | ip | IP address of a remote client  **Note**: Introduced in version 4.14.0. |
| SID | sid | Session ID of a remote client  **Note**: Introduced in version 4.14.0. |
| PHID | ncbi\_phid | CGI page hit ID  **Note**: Introduced in version 4.17.0 |

**Example**:

STAT QCLASSES

OK:[qclass my2]

OK:delete\_request: false

OK:timeout: 906

OK:notif\_hifreq\_interval: 0.1

OK:notif\_hifreq\_period: 5

OK:notif\_lofreq\_mult: 50

OK:dump\_buffer\_size: 100

OK:run\_timeout: 3600

OK:program\_name:

OK:failed\_retries: 0

OK:blacklist\_time: 0

OK:max\_input\_size: 2048

OK:max\_output\_size: 2048

OK:subm\_hosts:

OK:wnode\_hosts:

OK:wnode\_timeout: 40

OK:pending\_timeout: 604800

OK:max\_pending\_wait\_timeout: 0

OK:description:

OK:run\_timeout\_precision: 3600

OK:END

### The STAT QUEUES command

**Privileges**: any.

**Requires a queue**: yes

**NetSchedule output type**: multiple lines

**Version**: 4.14.0 and up

**Synopsis**:

STAT QUEUES [IP] [SID] [PHID]

**Description**: prints the queues with their parameters.

|  |  |  |
| --- | --- | --- |
| Parameter | Name | Description |
| IP | ip | IP address of a remote client  **Note**: Introduced in version 4.14.0. |
| SID | sid | Session ID of a remote client  **Note**: Introduced in version 4.14.0. |
| PHID | ncbi\_phid | CGI page hit ID  **Note**: Introduced in version 4.17.0 |

**Example**:

STAT QUEUES

OK:[queue CRASH10]

OK:kind: static

OK:position: 1

OK:qclass:

OK:delete\_request: false

OK:timeout: 3600

OK:notif\_hifreq\_interval: 0.1

OK:notif\_hifreq\_period: 5

OK:notif\_lofreq\_mult: 50

OK:dump\_buffer\_size: 100

OK:run\_timeout: 600

OK:program\_name:

OK:failed\_retries: 30

OK:blacklist\_time: 0

OK:max\_input\_size: 1000000

OK:max\_output\_size: 1000000

OK:subm\_hosts:

OK:wnode\_hosts:

OK:wnode\_timeout: 40

OK:pending\_timeout: 604800

OK:max\_pending\_wait\_timeout: 0

OK:description:

OK:run\_timeout\_precision: 30

OK:END

**Note:**

NS 4.17.0 and up adds ‘pause’ output parameter with values ‘nopause’, ‘nopullback’ or ‘pullback’.

### The STAT SERVICES command

**Privileges**: any.

**Requires a queue**: no

**NetSchedule output type**: single line

**Version**: 4.17.0 and up

**Synopsis**:

STAT SERVICES [IP] [SID] [PHID]

**Description**: prints the effective correspondences between services and queues.

|  |  |  |
| --- | --- | --- |
| Parameter | Name | Description |
| IP | ip | IP address of a remote client |
| SID | sid | Session ID of a remote client |
| PHID | ncbi\_phid | CGI page hit ID  **Note**: Introduced in version 4.17.0 |

**Example**:

STAT SERVICES

OK:AnotherService=TEST&MyServiceName=TEST

### The STAT ALERTS command

**Privileges**: any.

**Requires a queue**: no

**NetSchedule output type**: multiple lines

**Version**: 4.17.0 and up

**Synopsis**:

STAT ALERTS [IP] [SID] [PHID]

**Description**: prints the queues with their parameters.

|  |  |  |
| --- | --- | --- |
| Parameter | Name | Description |
| IP | Ip | IP address of a remote client |
| SID | Sid | Session ID of a remote client |
| PHID | ncbi\_phid | CGI page hit ID  **Note**: Introduced in version 4.17.0 |

**Example**:

STAT ALERTS

OK:[alert startaftercrash]

OK:last\_detected\_time: 03/19/2014 10:30:06.915799

OK:acknowledged\_time: n/a

OK:on: true

OK:count: 1

### The DUMP command

**Privileges**: any.

**Requires a queue**: yes

**NetSchedule output type**: multiple lines

**Version**: any / 4.10.0 and up (see parameters description)

**Synopsis**:

DUMP [[JobKey] |[ [Status] [StartAfter] [Count] [Group] [Affinity]]] [IP] [SID] [PHID] [Fields] [Order]

**Description**: prints a dump of a single job or multiple jobs. If a job key is provided then that job is dumped. Alternatively all or a certain number a jobs are dumped possibly starting from a certain job possibly restricting by a certain job state.

**Note**: starting from NS 4.25.0 the command is affected by the current scope (see SETSCOPE). An empty scope means that all the jobs will be included. The no-scope-only scope means that only the jobs which were submitted with an empty scope will be included. All the other scope identifiers mean that the only jobs from a particular scope will be included.

|  |  |  |
| --- | --- | --- |
| Parameter | Name | Description |
| JobKey | job\_key | String identifier: the job to dump. |
| Status | status | String identifier: dump only jobs which are in the given state.  The parameter is introduced in NetSchedule 4.10.0.  Version 4.20.0 introduces support of a list of statuses. The list is case insensitive and comma separated. |
| StartAfter | start\_after | String identifier: the job to start dumping after.  The parameter is introduced in NetSchedule 4.10.0. |
| Count | count | Integer: maximum number of jobs to dump. 0 => unlimited.  The parameter is introduced in NetSchedule 4.10.0. |
| Group | group | String identifier: if provided then only jobs from this group will be dumped.  The parameter is introduced in NetSchedule 4.10.0.  **Note**: Starting from NS 4.29.0 the ‘-‘ group could be used here. The ‘-‘ group is a functional equivalent of a no group. |
| Affinity | aff | String identifier: if provided then only jobs with this affinity will be dumped.  The parameter is introduced in NetSchedule 4.20.0.  **Note**: starting from NS 4.29.0 the ‘-‘ affinity could be used here. The ‘-‘ affinity is a functional equivalent of a no affinity. |
| IP | ip | IP address of a remote client  **Note**: Introduced in version 4.14.0. |
| SID | sid | Session ID of a remote client  **Note**: Introduced in version 4.14.0. |
| PHID | ncbi\_phid | CGI page hit ID  **Note**: Introduced in version 4.17.0 |
| Fields | fields | List of comma separated strings. If not provided or empty or only invalid names are in the list then all the job fields are provided.  Accepted fields: id, key, status, last\_touch, erase\_time, run\_expiration, read\_expiration, subm\_notif\_port, subm\_notif\_expiration, listener\_notif, listener\_notif\_expiration, events, run\_counter, read\_counter, affinity, group, mask, input, output, progress\_msg, remote\_client\_sid, remote\_client\_ip, ncbi\_phid, need\_subm\_progress\_msg\_notif, need\_lsnr\_progress\_msg\_notif, need\_stolen\_notif, gc\_erase\_time, scope. |
| Order | order | String identifier.  Accepted values are first and last. If last then the last (in terms of submit time) matching jobs are taken. |

**Example**:

DUMP JSID\_01\_1\_130.14.24.194\_9102

OK:id: 1

OK:key: JSID\_01\_1\_130.14.24.194\_9102

OK:status: Running

OK:erase\_time: n/a (duration 3600 sec)

OK:run\_expiration: 01/11/2012 11:06:00 (duration 1800 sec)

OK:read\_expiration: n/a (duration 1800 sec)

OK:subm\_notif\_port: n/a

OK:subm\_notif\_expiration: n/a

OK:event1: client=localhost event=Submit status=Pending ret\_code=0 timestamp='01/11/2012 10:04:19' node='' session='' err\_msg=''

OK:event2: client=localhost event=Request status=Running ret\_code=0 timestamp='01/11/2012 10:36:00' node='mwebdev34:7600' session='123456' err\_msg=''

OK:run\_counter: 1

OK:read\_counter: 0

OK:affinity: 1 ('99')

OK:mask: 0

OK:input: 'myinput'

OK:output: ''

OK:progress\_msg: ''

OK:remote\_client\_sid:

OK:remote\_client\_ip: 127.0.0.1

OK:END

### The AFLS command

**Privileges**: any.

**Requires a queue**: yes

**NetSchedule output type**: single line

**Version**: any / NetSchedule 4.10.0 has different output format

**Synopsis**:

AFLS [IP] [SID] [PHID]

**Description**: prints the number of jobs per affinity.

**Note**: starting from NS 4.29.0 there will be no jobs without an affinity. If a client submits a job without an affinity then the server silently assigns the ‘-‘ affinity to the job.

|  |  |  |
| --- | --- | --- |
| Parameter | Name | Description |
| IP | ip | IP address of a remote client  **Note**: Introduced in version 4.14.0. |
| SID | sid | Session ID of a remote client  **Note**: Introduced in version 4.14.0. |
| PHID | ncbi\_phid | CGI page hit ID  **Note**: Introduced in version 4.17.0 |

**Output synopsis 4.8.1 and 4.9.0**:

OK:<AffToken>=<JobsCount>[&<AffToken>=<JobsCount>…]

|  |  |
| --- | --- |
| Parameter | Description |
| AffToken | String: affinity token |
| JobsCount | Integer: number of jobs with this affinity regardless of their status. |

The only those affinities are printed which have non zero JobsCount.

**Output synopsis 4.10.0**:

OK:<AffToken>=<Count1>,<Count2>,<Count3>,<Count4>[&<AffToken>=<Count1>,…]

|  |  |
| --- | --- |
| Parameter | Description |
| AffToken | String: affinity token |
| Count1 | Integer: number of jobs in the “Pending” state with this affinity |
| Count2 | Integer: number of jobs in the “Running” state with this affinity. |
| Count3 | Integer: number of clients which have this affinity in their list of preferred affinities. |
| Count4 | Integer: number of clients which provided this affinity in the list of explicit affinities in the WGET/GET2 command and are still waiting for a job. |

All the known affinities are printed.

**Example for NetSchedule 4.8.1 and 4.9.0**:

AFLS

OK:aff1=14&aff23=4&other\_affinity=679

**Example for NetSchedule 4.10.0**:

AFLS

OK:aff1=6,8,2,0&aff23=4,2,0,0&other\_affinity=0,24,0,5

### The GETP command

**Privileges**: any.

**Requires a queue**: yes

**NetSchedule output type**: single line

**Synopsis**:

GETP [IP] [SID] [PHID]

**Description**: prints the queue max input and output sizes and the server features.

|  |  |  |
| --- | --- | --- |
| Parameter | Name | Description |
| IP | ip | IP address of a remote client  **Note**: Introduced in version 4.14.0. |
| SID | sid | Session ID of a remote client  **Note**: Introduced in version 4.14.0. |
| PHID | ncbi\_phid | CGI page hit ID  **Note**: Introduced in version 4.17.0 |

**Output synopsis**:

OK:max\_input\_size=<InputLimit>;max\_output\_size=<OutputLimit>;[<id1>=<val1>[;…]]

|  |  |
| --- | --- |
| Parameter | Description |
| InputLimit | Integer: input limit in bytes |
| OutputLimit | Integer: output limit in bytes |
| Id1 | Some NetSchedule feature identifier |
| Val1 | Some NetSchedule feature value |

**Example**:

GETP

OK:max\_input\_size=1048576;max\_output\_size=1048576;fast\_status=1;dyn\_queues=1;read\_confirm=1;version=0.0.0

### The GETP2 command

**Privileges**: any.

**Requires a queue**: yes

**NetSchedule output type**: single line

**Version**: 4.16.9 and up

**Synopsis**:

GETP2 [IP] [SID] [PHID]

**Description**: prints the queue max input and output sizes and, if so, NetCache linked section parameters. Each parameter from the NetCache section is prepended with ‘nc::’ prefix.

|  |  |  |
| --- | --- | --- |
| Parameter | Name | Description |
| IP | ip | IP address of a remote client  **Note**: Introduced in version 4.14.0. |
| SID | sid | Session ID of a remote client  **Note**: Introduced in version 4.14.0. |
| PHID | ncbi\_phid | CGI page hit ID  **Note**: Introduced in version 4.17.0 |

**Output synopsis**:

OK:max\_input\_size=<InputLimit>&max\_output\_size=<OutputLimit>&[<id1>=<val1>[&…]]

|  |  |
| --- | --- |
| Parameter | Description |
| InputLimit | Integer: input limit in bytes |
| OutputLimit | Integer: output limit in bytes |
| Id1 | value identifier |
| Val1 | url encoded value |

**Example**:

GETP2

OK:max\_input\_size=1048576&max\_output\_size=1048576&nc::service=NC\_Test

### The GETC command

**Privileges**: any.

**Requires a queue**: yes

**NetSchedule output type**: multiple lines

**Synopsis**:

GETC [IP] [SID] [PHID]

**Description**: prints the queue parameters.

|  |  |  |
| --- | --- | --- |
| Parameter | Name | Description |
| IP | ip | IP address of a remote client  **Note**: Introduced in version 4.14.0. |
| SID | sid | Session ID of a remote client  **Note**: Introduced in version 4.14.0. |
| PHID | ncbi\_phid | CGI page hit ID  **Note**: Introduced in version 4.17.0 |

**Output synopsis (single line)**:

OK:<name>=<value>

|  |  |
| --- | --- |
| Parameter | Description |
| Name | The queue parameter name |
| value | The queue parameter value |

**Example**:

GETC

OK:timeout=3600

OK:notif\_hifreq\_interval=0.1

OK:notif\_hifreq\_period=5

OK:notif\_lofreq\_mult=50

OK:dump\_buffer\_size=100

OK:run\_timeout=1800

OK:run\_timeout\_precision=30

OK:failed\_retries=0

OK:blacklist\_time=0

OK:empty\_lifetime=-1

OK:max\_input\_size=1048576

OK:max\_output\_size=1048576

OK:deny\_access\_violations=false

OK:program=

OK:subm\_host=

OK:wnode\_host=

OK:END

### The CANCELQ command

**Privileges**: any.

**Requires a queue**: yes

**NetSchedule output type**: single line

**Version**: 4.10.0 and up

**Synopsis**:

CANCELQ [IP] [SID] [PHID]

**Description**: cancels all the jobs in the queue. Please note that the command execution time depends linear from the number of jobs in the queue so the execution can be very long. While the command is executed the other clients cannot perform any operations with the queue.

**Note**: starting from NS 4.25.0 the command is affected by the current scope (see SETSCOPE). The scope restricts the candidate jobs. Both, empty scope identifier and the no-scope-only identifier are treated as the candidate jobs must not belong to any scope.

|  |  |  |
| --- | --- | --- |
| Parameter | Name | Description |
| IP | ip | IP address of a remote client  **Note**: Introduced in version 4.14.0. |
| SID | sid | Session ID of a remote client  **Note**: Introduced in version 4.14.0. |
| PHID | ncbi\_phid | CGI page hit ID  **Note**: Introduced in version 4.17.0 |

**Output synopsis** (NS below 4.20.0):

OK:

**Output synopsis** (NS 4.20.0 and up):

OK:<number of canceled jobs>

### The REFUSESUBMITS command

**Privileges**: admin.

**Requires a queue**: see the description.

**NetSchedule output type**: single line

**Version**: 4.11.0 and up

**Synopsis**:

REFUSESUBMITS <Mode> [IP] [SID] [PHID]

|  |  |  |
| --- | --- | --- |
| Parameter | Name | Description |
| Mode | mode | Integer value: 0 or 1. |
| IP | ip | IP address of a remote client  **Note**: Introduced in version 4.14.0. |
| SID | sid | Session ID of a remote client  **Note**: Introduced in version 4.14.0. |
| PHID | ncbi\_phid | CGI page hit ID  **Note**: Introduced in version 4.17.0 |

**Description**: sets the refuse submits mode. The mode can be set on the server level or on a certain queue level separately. If a queue name is not given on the handshake stage then the command affects the server level mode. If the queue name is provided on the handshake stage then the command affects the queue mode. The SUBMIT or BSUB commands are refused by NetSchedule if the refuse submits mode is set to 1 on either the server or the queue level.

**Output synopsis**:

OK:

### The QPAUSE command

**Privileges**: none

**Requires a queue**: yes

**NetSchedule output type**: single line

**Version**: 4.17.0 and up

**Synopsis**:

QPAUSE <Pullback> [IP] [SID] [PHID]

|  |  |  |
| --- | --- | --- |
| Parameter | Name | Description |
| Pullback | pullback | Integer value: 0 or 1. Default is 0.  If pullback is 0 then the value of the ‘pause’ parameter in the SST2, WST2, STATUS2, GET2 commands will be ‘nopullback’  If pullback is 1 then the value of the ‘pause’ parameter in the SST2, WST2, STATUS2, GET2 commands will be ‘pullback’ |
| IP | ip | IP address of a remote client |
| SID | sid | Session ID of a remote client |
| PHID | ncbi\_phid | CGI page hit ID  **Note**: Introduced in version 4.17.0 |

**Description**: sets the pause mode. When the queue is paused no jobs will be given to worker nodes when they request one (GET, WGET, GET2 commands).

**Output synopsis**:

OK:

### The QRESUME command

**Privileges**: none

**Requires a queue**: yes

**NetSchedule output type**: single line

**Version**: 4.17.0 and up

**Synopsis**:

QRESUME [IP] [SID] [PHID]

|  |  |  |
| --- | --- | --- |
| Parameter | Name | Description |
| IP | Ip | IP address of a remote client |
| SID | sid | Session ID of a remote client |
| PHID | ncbi\_phid | CGI page hit ID  **Note**: Introduced in version 4.17.0 |

**Description**: clears the pause mode.

**Output synopsis**:

OK:

### The SETQUEUE command

**Privileges**: none

**Requires a queue**: see the description

**NetSchedule output type**: single line

**Version**: 4.12.0 and up

**Synopsis**:

SETQUEUE [QueueName] [IP] [SID] [PHID]

|  |  |  |
| --- | --- | --- |
| Parameter | Name | Description |
| QueueName | qname | String queue identifier |
| IP | ip | IP address of a remote client  **Note**: Introduced in version 4.14.0. |
| SID | sid | Session ID of a remote client  **Note**: Introduced in version 4.14.0. |
| PHID | ncbi\_phid | CGI page hit ID  **Note**: Introduced in version 4.17.0 |

**Description**: Switches between queues without necessity to reconnect. If no queue name is given or a predefined ‘noname’ is used then the current queue is reset.

**Output synopsis**:

OK:

## Submitter / Worker Node / Reader Common Commands

### The QUIT command

**Privileges**: any.

**Requires a queue**: no

**NetSchedule output type**: no output

**Synopsis**:

QUIT [IP] [SID] [PHID]

**Description**: closes the connection.

|  |  |  |
| --- | --- | --- |
| Parameter | Name | Description |
| IP | ip | IP address of a remote client  **Note**: Introduced in version 4.14.0. |
| SID | sid | Session ID of a remote client  **Note**: Introduced in version 4.14.0. |
| PHID | ncbi\_phid | CGI page hit ID  **Note**: Introduced in version 4.17.0 |

## Submitter Commands

### The MGET command

**Privileges**: any.

**Requires a queue**: yes

**NetSchedule output type**: single line

**Synopsis**:

MGET<JobKey> [IP] [SID] [PHID]

**Description**: Prints the progress message for a job.

|  |  |  |
| --- | --- | --- |
| Parameter | Name | Description |
| JobKey | job\_key | String identifier: the job key. |
| IP | ip | IP address of a remote client  **Note**: Introduced in version 4.14.0. |
| SID | sid | Session ID of a remote client  **Note**: Introduced in version 4.14.0. |
| PHID | ncbi\_phid | CGI page hit ID  **Note**: Introduced in version 4.17.0 |

**Output synopsis**:

OK:<Message>

|  |  |
| --- | --- |
| Parameter | Description |
| Message | String: the progress message for the job. The message is printed without double quotes whether or not there were any space characters in the message. |

### The SST command

**Privileges**: any.

**Requires a queue**: yes

**NetSchedule output type**: single line

**Synopsis**:

SST<JobKey> [IP] [SID] [PHID]

**Description**: prints the state of a job. The command is obsolete starting from NetSchedule 4.10.0, use SST2 instead.

|  |  |  |
| --- | --- | --- |
| Parameter | Name | Description |
| JobKey | job\_key | String identifier: the job key. |
| IP | ip | IP address of a remote client  **Note**: Introduced in version 4.14.0. |
| SID | sid | Session ID of a remote client  **Note**: Introduced in version 4.14.0. |
| PHID | ncbi\_phid | CGI page hit ID  **Note**: Introduced in version 4.17.0 |

**Output synopsis**:

OK:<State>

|  |  |
| --- | --- |
| Parameter | Description |
| State | Integer: the state of the job  0 – pending  1 – running  3 – canceled  4 – failed  5 – done  6 – reading  7 – confirmed  8 – read failed  -1 – job not found |

**Example**:

SST JSID\_01\_4\_130.14.24.194\_9102

OK:0

### The SST2 command

**Privileges**: any.

**Requires a queue**: yes

**NetSchedule output type**: single line

**Version**: 4.10.0 and up, see also description of the output parameters

**Synopsis**:

SST2<JobKey> [IP] [SID] [PHID] [NeedProgressMessage]

**Description**: prints the state of a job.

|  |  |  |
| --- | --- | --- |
| Parameter | Name | Description |
| JobKey | job\_key | String identifier: the job key. |
| IP | ip | IP address of a remote client  **Note**: Introduced in version 4.14.0. |
| SID | sid | Session ID of a remote client  **Note**: Introduced in version 4.14.0. |
| PHID | ncbi\_phid | CGI page hit ID  **Note**: Introduced in version 4.17.0 |
| NeedProgressMessage | need\_progress\_msg | Tells if the command output should also have the job progress message. The accepted values are 0 and 1. Default is 0. If set to 1 then the output will have the ‘msg’ parameter, see below.  **Note:** Introduces in version 4.31.0 |

**Output synopsis**:

OK:job\_status=<State>&job\_exptime=<AbsTime>[&pause=<PauseStatus>[&msg=<ProgressMessage]]

|  |  |
| --- | --- |
| Parameter | Description |
| State | The state of the job. It could be as a string:  Pending  Running  Canceled  Failed  Done  Reading  Confirmed  ReadFailed |
| AbsTime | **Note**: available for NS 4.11.0 and up  Time in seconds when a job is deleted if nothing else happens to the job. The time is an absolute UNIX time.  For the jobs in the Running and Pending states this value is calculated as the current time plus the queue timeout. |
| PauseStatus | **Note**: available for NS 4.17.0 and up  The parameter appears only of the corresponding queue is paused some way. The values are:  pullback  nopullback |
| ProgressMessage | **Note:** available for NS 4.31.0 and up.  The parameter appears only if the command has NeedProgressMessage parameter set to 1.  The value of the “ProgressMessage” is a URL encoded string. |

**Note**: if there is no such a job, NS 4.11.0 output will be as follows:

ERR:eJobNotFound:

**Example**:

SST2 JSID\_01\_4\_130.14.24.194\_9102

OK:job\_status=Running

### The SUBMIT command

**Privileges**: any.

**Requires a queue**: yes

**Version**: parameters differ depending on the version. See the description below.

**NetSchedule output type**: single line

**Synopsis**:

SUBMIT<Input> [ProgressMsg] [Port] [Timeout] [Aff] [Mask] [IP] [SID] [Group] [PHID] [NeedProgressMessage]

**Description**: submits a job.

**Note**: Starting from NS 4.25.0 the command is affected by the current scope (see SETSCOPE). Both, empty scope identifier and the no-scope-only identifier are treated as submitted job does not have a scope.

**Note**: starting from NS 4.29.0 if no affinity is specified then the server silently assigns the ‘-‘ affinity to the job.

**Note**: starting from NS 4.29.0 if no group is specified then the server silently assigns the ‘-‘ group to the job.

|  |  |  |
| --- | --- | --- |
| Parameter | Name | Description |
| Input | input | The job input. |
| ProgressMsg | progress\_msg | The job progress message.  Not supported in NetSchedule 4.10.0 and up. |
| Port | port | The client port number on which the client expects notifications about changes in the submitted job states. |
| Timeout | timeout | The timeout within which the client expects notifications about changes in the submitted job states. |
| Mask | msk | The job mask.  The value is not used for making any decisions. The parameter is for future extensions. |
| IP | ip | The remote client IP |
| SID | sid | The remote client session ID |
| Group | group | String identifier: if given then the job will be included into this group.  The parameter is introduced in NetSchedule 4.10.0  (See note above for NS 4.29.0 and up) |
| Aff | aff | The job affinity identifier.  The allowed set of symbols is [a-z][A-Z][0-9]\_  (See note above for NS 4.29.0 and up) |
| PHID | ncbi\_phid | CGI page hit ID  **Note**: Introduced in version 4.17.0 |
| NeedProgressMessage | need\_progress\_message | It tells if the submitter wants to receive the notifications when the job progress message is changed. Valid values are 0 and 1. Default is 0. If set to 1 then the notifications will be sent.  **Note:** introduced in version 4.31.0 |

**Output synopsis**:

OK:<JobKey>

|  |  |
| --- | --- |
| Parameter | Description |
| JobKey | The job key NetSchedule issued for the submitted job. |

**Example**:

SUBMIT bla

OK:JSID\_01\_5\_130.14.24.194\_9102

### The LISTEN command

**Privileges**: any.

**Requires a queue**: yes

**Version**: NetSchedule 4.14.0 and up

**NetSchedule output type**: single line

**Synopsis**:

LISTEN <JobKey> <Port> <Timeout> [IP] [SID] [PHID] [NeedStolen] [NeedProgressMessage]

**Description**: Sets the listener for job state change notifications. The address for the notifications is taken from the client connection. NetSchedule supports one listener per job. Any consequent LISTEN command for a job overwrites the previous values. To reset listening, the port or timeout parameters should be given as 0. The notifications caused by this command are sent when a job changes its state to any other state, i.e. any state transition will be accompanied by the corresponding notification. Starting from NS 4.31.0 a client may request a notification in case if another client stole the notification host:port for itself. Also NS 4.31.0 lets to specify if notifications in case of the changed progress message should be sent.

|  |  |  |
| --- | --- | --- |
| Parameter | Name | Description |
| JobKey | job\_key | Job key.  The job to listen to state change notifications. |
| Port | port | The client port number on which the client expects notifications about changes in the job states. |
| Timeout | timeout | The timeout in seconds within which the client expects notifications about changes in the job states. |
| IP | ip | IP address of a remote client  **Note**: Introduced in version 4.14.0. |
| SID | sid | Session ID of a remote client  **Note**: Introduced in version 4.14.0. |
| PHID | ncbi\_phid | CGI page hit ID  **Note**: Introduced in version 4.17.0 |
| NeedStolen | need\_stolen | It tells if a notification should be sent in case if another client steals the host:port for the notifications. The accepted values are 0 and 1. Default is 0. If set to 1 then the stolen notifications will be sent.  **Note:** introduced in version 4.31.0 |
| NeedProgressMessage | need\_progress\_msg | It tells if a notification should be sent in case if a job progress message was updated. The accepted values are 0 and 1. Default is 0. If set to 1 then the progress message changed notifications will be sent.  Also it affects the LISTEN command response. If set to 1 then the output will have the ‘msg’ parameter with the URL encoded progress message.  **Note:** introduced in version 4.31.0. |

**Output synopsis**:

OK:job\_status=<JobState>&last\_event\_index=<EventIndex>[&msg=<ProgressMessage>]

|  |  |
| --- | --- |
| Parameter | Description |
| JobState | The current job state. |
| EventIndex | Integer; the index of the last event in a job event history. For example if a job has just been submitted its status will be Pending and there is exactly one event in a job history so the provided value will be 0. |
| ProgressMessage | If the LISTEN command has the NeedProgressMessage value set to 1 then this parameter will be included into the output. The value is the URL encoded job progress message.  **Note:** introduced in version 4.31.0. |

In case if the given job does not exist the output will be as follows:

ERR:eJobNotFound:

**Example**:

LISTEN JSID\_01\_5\_130.14.24.194\_9102 5790 300

OK:job\_status=Running&last\_event\_index=1

### The CANCEL command

**Privileges**: any.

**Requires a queue**: yes

**NetSchedule output type**: single line

**Synopsis**:

CANCEL <JobKey | ( Group | Affinity | Status | <any combination of Group, Affinity, Status> )> [IP] [SID] [PHID]

**Description**: Cancels the job or all the jobs within the given group or all jobs with the given affinity or all jobs in the given group with the given affinity. If a job key is provided then neither group nor affinity must be given.

**Note**: starting from NS 4.25.0 the command is affected by the current scope (see SETSCOPE). The scope restricts the candidate jobs. Both, empty scope identifier and the no-scope-only identifier are treated as the candidate job must not belong to any scope.

|  |  |  |
| --- | --- | --- |
| Parameter | Name | Description |
| JobKey | job\_key | String identifier: the job key. |
| Group | group | String identifier: the jobs group.  The parameter is introduced in NetSchedule 4.10.0  **Note**: Starting from NS 4.29.0 the ‘-‘ group could be provided here. It is a functional equivalent of a no group job. |
| Affinity | aff | String identifier: the affinity identifier  The parameter is introduced in NetSchedule 4.17.2  **Note**: Starting from NS 4.29.0 the ‘-‘ affinity could be provided here. It is a functional equivalent of a no affinity job. |
| Status | status | String: a list of comma separated job statuses which should be canceled. The list is not case sensitive. If the ‘canceled’ is found then a warning is produced. Another warning is produced if a status is found more than once in the list.  **Note**: introduced in version 4.20.0 |
| IP | ip | IP address of a remote client  **Note**: Introduced in version 4.14.0. |
| SID | sid | Session ID of a remote client  **Note**: Introduced in version 4.14.0. |
| PHID | ncbi\_phid | CGI page hit ID  **Note**: Introduced in version 4.17.0 |

**Output synopsis** (NS below 4.20.0):

OK:

**Output** **synopsis** (NS 4.20.0 and up):

OK:<number of canceled jobs>

### The BSUB command

**Privileges**: any.

**Requires a queue**: yes

The BSUB command (batch submit) is a multi-stage command. The overall structure is shown on the figure below.



**Description**: the batch submit starts with the BSUB command. Then an arbitrary number of batches may appear. And then the batch submit is completed by the ENDS command.

Each batch starts with the BTCH command followed by individual lines with job parameters (one line describes one job) and then followed by the ENDS command.

**Note**: Starting from NS 4.25.0 the command is affected by the current scope (see SETSCOPE). Both, empty scope identifier and the no-scope-only identifier are treated as submitted job does not have a scope.

**Note**: Starting from NS 4.29.0 if no group is supplied then the server silently uses the ‘-‘ group.

**BSUB Synopsis**:

BSUB [Port] [Timeout] [IP] [SID] [Group] [PHID]

|  |  |  |
| --- | --- | --- |
| Parameter | Name | Description |
| Port | port | The client port number on which the client expects notifications about changes in the submitted jobs states. |
| Timeout | timeout | The timeout within which the client expects notifications about changes in the submitted jobs states. |
| IP | ip | The remote client IP |
| SID | sid | The remote client session ID |
| Group | group | If given then jobs within the batch submit will be included into this group. The group is a string identifier.  The parameter is introduced in NetSchedule 4.10.0.  (see note for NS 4.29.0 above) |
| PHID | ncbi\_phid | CGI page hit ID  **Note**: Introduced in version 4.17.0 |

**BSUB output synopsis**:

OK:Batch submit ready

**ENDS Synopsis**:

ENDS

**ENDS output synopsis**:

OK:

**BTCH Synopsis**:

BTCH <Size>

|  |  |  |
| --- | --- | --- |
| Parameter | Name | Description |
| Size | Size | Integer: the number of jobs expected in the batch, i.e. the number of lines with individual job parameters. |

**BTCH output synopsis**:

No output will be provided

**Individual job parameters line synopsis**:

<Input> [Aff] [Mask]

|  |  |  |
| --- | --- | --- |
| Parameter | Name | Description |
| Input | input | The job input. |
| Aff | aff | The job affinity identifier.  The allowed set of symbols is [a-z][A-Z][0-9]\_  **Note**: starting from NS 4.29.0 if no affinity is supplied then the server silently uses the ‘-‘ affinity for the job. |
| Mask | msk | The job mask.  The value is not used for making any decisions. The parameter is for future extensions. |

**Individual job parameters line output synopsis**:

No output will be provided

**ENDB Synopsis**:

ENDB

**ENDB output synopsis**:

OK:<StartID> <HostIP> <Port>

|  |  |
| --- | --- |
| Parameter | Description |
| StartID | The integer identifier of the first submitted job within the batch. It is guaranteed that the job ids will be monotonically increased for all the jobs in the batch. |
| HostIP | The NetSchedule host IP address |
| Port | The NetSchedule listening port |

## Worker Node Commands

### The MPUT command

**Privileges**: any.

**Requires a queue**: yes

**NetSchedule output type**: single line

**Synopsis**:

MPUT <JobKey> <Message> [IP] [SID] [PHID]

**Description**: Associates a progress message with a job.

|  |  |  |
| --- | --- | --- |
| Parameter | Name | Description |
| JobKey | job\_key | String identifier: the job key. |
| Message | progress\_msg | Message to be assigned. If the message has spaces in it, the message should put into double quotes. |
| IP | ip | IP address of a remote client  **Note**: Introduced in version 4.14.0. |
| SID | sid | Session ID of a remote client  **Note**: Introduced in version 4.14.0. |
| PHID | ncbi\_phid | CGI page hit ID  **Note**: Introduced in version 4.17.0 |

### The CLRN command

**Privileges**: any.

**Requires a queue**: yes

**NetSchedule output type**: single line

**Version**: the description is given for NetSchedule 4.10.0 and up. The previous versions had another set of parameters and for them the command is silently ignored.

**Synopsis**:

CLRN [IP] [SID] [PHID]

**Description**: Clears information about the client from the client registry. If the client had any running jobs they will be moved to the pending state. If the client had any reading jobs they will be moved to the done state. The client is expected to be identified, i.e. the client\_node and client\_session parameters are provided at the handshaking phase.

|  |  |  |
| --- | --- | --- |
| Parameter | Name | Description |
| IP | ip | IP address of a remote client  **Note**: Introduced in version 4.14.0. |
| SID | sid | Session ID of a remote client  **Note**: Introduced in version 4.14.0. |
| PHID | ncbi\_phid | CGI page hit ID  **Note**: Introduced in version 4.17.0 |

**Output synopsis**:

OK:

### The WST command

**Privileges**: any.

**Requires a queue**: yes

**NetSchedule output type**: single line

**Synopsis**:

WST<JobKey> [IP] [SID] [PHID]

**Description**: prints the job state. The command is obsolete starting from NetSchedule 4.10.0, use WST2 instead.

|  |  |  |
| --- | --- | --- |
| Parameter | Name | Description |
| JobKey | job\_key | String identifier: the job key. |
| IP | ip | IP address of a remote client  **Note**: Introduced in version 4.14.0. |
| SID | sid | Session ID of a remote client  **Note**: Introduced in version 4.14.0. |
| PHID | ncbi\_phid | CGI page hit ID  **Note**: Introduced in version 4.17.0 |

**Output synopsis**:

OK:job\_status=<State>

|  |  |
| --- | --- |
| Parameter | Description |
| State | Integer: the job state  0 – pending  1 – running  3 – canceled  4 – failed  5 – done  6 – reading  7 – confirmed  8 – read failed  -1 – not found |

**Example**:

WST JSID\_01\_4\_130.14.24.194\_9102

OK:0

### The WST2 command

**Privileges**: any.

**Requires a queue**: yes

**NetSchedule output type**: single line

**Version**: NetSchedule 4.10.0 and up, see also description of the output parameters

**Synopsis**:

WST2<JobKey> [IP] [SID] [PHID] [NeedProgressMessage]

**Description**: prints the job state.

|  |  |  |
| --- | --- | --- |
| Parameter | Name | Description |
| JobKey | job\_key | String identifier: the job key. |
| IP | ip | IP address of a remote client  **Note**: Introduced in version 4.14.0. |
| SID | sid | Session ID of a remote client  **Note**: Introduced in version 4.14.0. |
| PHID | ncbi\_phid | CGI page hit ID  **Note**: Introduced in version 4.17.0 |
| NeedProgressMessage | need\_progress\_msg | Tells if the command output should also have the job progress message. The accepted values are 0 and 1. Default is 0. If set to 1 then the output will have the ‘msg’ parameter, see below.  **Note:** Introduces in version 4.31.0 |

**Output synopsis**:

OK:job\_status=<State>&job\_exptime=<AbsTime>[&pause=<PauseStatus>[&msg=<ProgressMessage]]

|  |  |
| --- | --- |
| Parameter | Description |
| State | The job state as a string. It could be one of the following:  Pending  Running  Canceled  Failed  Done  Reading  Confirmed  ReadFailed |
| AbsTime | **Note**: available for NS 4.11.0 and up  Time in seconds when a job is deleted if nothing else happens to the job. The time is an absolute UNIX time.  For the jobs in the Running and Pending states this value is calculated as the current time plus the queue timeout. |
| PauseStatus | **Note**: available for NS 4.17.0 and up  The parameter appears only of the corresponding queue is paused some way. The values are:  pullback  nopullback |
| ProgressMessage | **Note:** available for NS 4.31.0 and up.  The parameter appears only if the command has NeedProgressMessage parameter set to 1.  The value of the “ProgressMessage” is a URL encoded string. |

**Note**: if there is no such a job, NS 4.11.0 output will be as follows:

ERR:eJobNotFound:

**Example**:

WST2 JSID\_01\_4\_130.14.24.194\_9102

OK:job\_status=Canceled

### The CHAFF command

**Privileges**: any.

**Requires a queue**: yes

**NetSchedule output type**: single line

**Version**: 4.10.0 and up

**Synopsis**:

CHAFF [AffToAdd] [AffToDel] [IP] [SID] [PHID]

**Description**: informs NetSchedule server about the changes in the client list of preferred affinities. The client must be an identified one, i.e. the client\_node and client\_session parameters must be provided in the handshaking phase.

|  |  |  |
| --- | --- | --- |
| Parameter | Name | Description |
| AffToAdd | add | Comma or tab separated list of affinity identifiers to be added into the list of the client preferred affinities.  The allowed set of symbols for each affinity identifier is [a-z][A-Z][0-9]\_  **Note**: Starting from NS 4.29.0 if the ‘-‘ affinity is supplied then it is silently ignored. |
| AffToDel | del | Comma or tab separated list of affinity identifiers to be removed from the list of the client preferred affinities.  The allowed set of symbols for each affinity identifier is [a-z][A-Z][0-9]\_  **Note**: Starting from NS 4.29.0 if the ‘-‘ affinity is supplied then it is silently ignored. |
| IP | ip | IP address of a remote client  **Note**: Introduced in version 4.14.0. |
| SID | sid | Session ID of a remote client  **Note**: Introduced in version 4.14.0. |
| PHID | ncbi\_phid | CGI page hit ID  **Note**: Introduced in version 4.17.0 |

**Output synopsis**:

OK:

**Example**:

CHAFF a1,a2

OK:

### The SETAFF command

**Privileges**: any.

**Requires a queue**: yes

**NetSchedule output type**: single line

**Version**: 4.16.4 and up

**Synopsis**:

SETAFF [AffToSet] [IP] [SID] [PHID]

**Description**: informs NetSchedule server about what the client list of preferred affinities is. The client must be an identified one, i.e. the client\_node and client\_session parameters must be provided in the handshaking phase.

|  |  |  |
| --- | --- | --- |
| Parameter | Name | Description |
| AffToSet | aff | Comma or tab separated list of affinity identifiers to be set as the list of the client preferred affinities. The list can be empty.  The allowed set of symbols for each affinity identifier is [a-z][A-Z][0-9]\_  **Note**: Starting from NS 4.29.0 if the ‘-‘ affinity is supplied then it is silently ignored. |
| IP | ip | IP address of a remote client |
| SID | sid | Session ID of a remote client |
| PHID | ncbi\_phid | CGI page hit ID  **Note**: Introduced in version 4.17.0 |

**Output synopsis**:

OK:

**Example**:

SETAFF a1,a2

OK:

### The GET command

**Privileges**: any.

**Requires a queue**: yes

**NetSchedule output type**: single line

**Version**: it is obsolete starting from NetSchedule 4.10.0. Use GET2 instead.

**Synopsis**:

GET [Port] [Aff] [IP] [SID] [PHID]

**Description**: provides a job for running. The command is becoming obsolete. Use GET2 instead.

**Note:** starting from NS 4.17.0 a queue could be paused (see QPAUSE command). If the queue is paused then no job will be given if there are some jobs which match the requested criteria.

**Note**: starting from NS 4.25.0 the command is affected by the current scope (see SETSCOPE). The scope restricts the candidate jobs. Both, empty scope identifier and the no-scope-only identifier are treated as the candidate job must not belong to any scope.

|  |  |  |
| --- | --- | --- |
| Parameter | Name | Description |
| Port | port | Integer: the port number to identify what notifications about vacant job availability is to be deleted from the list of notifications. |
| Aff | aff | A comma separated list of affinity identifiers.  If given then NetSchedule tries to pick a job with the mentioned affinities. This is the first priority of picking a job.  The allowed set of symbols for each identifier is [a-z][A-Z][0-9]\_  **Note**: starting from NS 4.29.0 the ‘-‘ affinity could be used here. It is a functional equivalent of no affinity jobs. |
| IP | ip | IP address of a remote client  **Note**: Introduced in version 4.14.0. |
| SID | sid | Session ID of a remote client  **Note**: Introduced in version 4.14.0. |
| PHID | ncbi\_phid | CGI page hit ID  **Note**: Introduced in version 4.17.0 |

**Output synopsis (suitable job found)**:

OK:<JobKey> <“Input”> <“Aff”> <“ClientIP ClientSession”> <Mask>

**Output synopsis (no suitable job found)**:

OK:

|  |  |
| --- | --- |
| Parameter | Description |
| JobKey | String identifier: the job key. |
| Input | String: the job input. |
| Aff | String: the job affinity identifier. |
| ClientIP | String: the remote client IP. |
| ClientSession | String: the remote client session. |
| Mask | Integer: the job mask. |

**Example**:

GET

OK:JSID\_01\_6\_130.14.24.194\_9102 "myinput" "a2" "127.0.0.1 " 0

### The GET2 command

**Privileges**: any.

**Requires a queue**: yes

**NetSchedule output type**: single line

**Version**: 4.10.0; 4.11.0 and up have another output format

**Synopsis**:

GET2 <WnodeAff> <AnyAff> [ExclusiveNewAff] [Aff] [Port] [Timeout] [Group] [IP] [SID] [PHID] [PrioritizedAff]

**Description**: provides a job for running. The client must be identified, i.e. the client\_node and client\_session parameters must be provided in the handshaking phase.

**Note:** starting from NS 4.17.0 a queue could be paused (see QPAUSE command). If the queue is paused then no job will be given if there are some jobs which match the requested criteria.

**Note**: starting from NS 4.25.0 the command is affected by the current scope (see SETSCOPE). The scope restricts the candidate jobs. Both, empty scope identifier and the no-scope-only identifier are treated as the candidate job must not belong to any scope.

|  |  |  |
| --- | --- | --- |
| Parameter | Name | Description |
| Port | Port | Integer: the port number on which the client will wait for a notification about vacant job availability if this GET command did not pick a job. |
| Timeout | timeout | Integer: the timeout within which the client will wait for a notification about vacant job availability if this GET command did not pick a job. |
| Aff | Aff | A comma separated list of affinity identifiers.  If given then NetSchedule tries to pick a job with the mentioned affinities. This is the first priority of picking a job.  The allowed set of symbols for each identifier is [a-z][A-Z][0-9]\_  **Note**: starting from NS 4.29.0 the ‘-‘ affinity could be used here. It is a functional equivalent of no affinity jobs. |
| WnodeAff | wnode\_aff | Integer: 0 or 1. If given then NetSchedule tries to pick a job bearing in mind the client preferred affinities list. This is the second priority of picking a job.  Introduced in NetSchedule 4.10.0.  Applicable only for identified clients. |
| AnyAff | any\_aff | Integer: 0 or 1. If 1 is given then any pending job will be picked regardless of an affinity. This is the third priority of picking a job.  Introduced in NetSchedule 4.10.0. |
| ExclusiveNewAff | exclusive\_new\_aff | Integer: 0 or 1. If set to 1 then NetSchedule tries to pick a job with no affinities or with an affinity which is not in the list of preferred affinities of all worker nodes i.e. exclusive new affinity. If the picked job has an affinity then it is added to the list of preferred affinities of the worker node.  This flag is allowed to set to 1 only if any\_aff is set to 0.  Introduced in NetSchedule 4.11.0. |
| Group | group | String identifier. If given as non-empty string then it is used as a restriction on what jobs could be provided for the worker node. If non-existing group is given then no errors is generated and no jobs will be given.  **Note**: if a group is given then jobs without a group will be excluded from the list of candidates.  **Note**: Starting from NetSchedule 4.23.0 a list of groups could be provided as a comma separated list.  **Note**: starting from NS 4.29.0 the ‘-‘ group could be used here. It is a functional equivalent of no group jobs. |
| IP | ip | IP address of a remote client  **Note**: Introduced in version 4.14.0. |
| SID | sid | Session ID of a remote client  **Note**: Introduced in version 4.14.0. |
| PHID | ncbi\_phid | CGI page hit ID  **Note**: Introduced in version 4.17.0 |
| PrioritizedAff | prioritized\_aff | Integer: 0 or 1. Could be set to 1 only if wnode\_aff, any\_aff and exclusive\_new\_aff are set to 0 and there is at least one affinity in the explicit affinity list.  If set to 1 then affinity in the explicit list are treated prioritized (first appeared has the highest priority) and a job is picked accordingly.  **Note**: introduced in version 4.22.0  **Note**: NS 4.27.0 and up let to have any\_aff set to 1 together with prioritized\_aff set to 1. If so then a job with any affinity (including no affinity) could be provided as the last resort. |

**Output synopsis (NetSchedule 4.10.0, suitable job found)**:

OK:<JobKey> <“Input”> <“Aff”> <“ClientIP ClientSession”> <Mask> <SecurityToken>

**Output synopsis (NetSchedule 4.11.0 and up, suitable job found)**:

OK:job\_key=<JobKey>&input=<Input>&affinity=<Aff>&client\_ip=<ClientIP>&client\_sid=<ClientSession>&mask=<Mask>&auth\_token=<SecurityToken>&ncbi\_phid=<PHID>

**Output synopsis (NetSchedule 4.30.0, suitable job found and the submitter requested notifications)**:

OK:job\_key=<JobKey>&input=<Input>&affinity=<Aff>&client\_ip=<ClientIP>&client\_sid=<ClientSession>&mask=<Mask>&auth\_token=<SecurityToken>&ncbi\_phid=<PHID>&submitter\_notif\_host=<ip>&submitter\_notif\_port=<port>

**Output synopsis (no suitable job found)**:

OK:

**Output synopsis (NetSchedule 4.17.0 and up, queue is paused):**

OK:pause=<PauseStatus>

|  |  |
| --- | --- |
| Parameter | Description |
| JobKey | String identifier: the job key. |
| Input | String: the job input.  URL encoded for 4.11.0 and up |
| Aff | String: the job affinity identifier.  URL encoded for 4.11.0 and up |
| ClientIP | String: the remote client IP.  URL encoded for 4.11.0 and up |
| ClientSession | String: the remote client session.  URL encoded for 4.11.0 and up |
| Mask | Integer: the job mask. |
| SecurityToken | String identifier: the job security token which must be used for putting, returning or failing the job. |
| PauseStatus | NS 4.17.0 and up  String, one of the following:  pullback  nopullback |
| PHID | CGI page hit ID which was associated with the job at the submit time.  Introduced for NS 4.17.0 |
| ip | Job submitter notification IP address.  The parameter and its value appears only if the submitter requested notifications. **Note**: the notification timeout is not taken into consideration.  Introduced in NS 4.30.0 |
| port | Job submitter notification port.  The parameter and its value appears only if the submitter requested notifications. **Note**: the notification timeout is not taken into consideration.  Introduced in NS 4.30.0 |

**Example**:

GET2 wnode\_aff=1 any\_aff=0

OK:JSID\_01\_6\_130.14.24.194\_9102 "input" "a2" "127.0.0.1 " 0 8764505\_3

### The PUT command

**Privileges**: any.

**Requires a queue**: yes

**NetSchedule output type**: single line

**Synopsis**:

PUT <JobKey> <RetCode> <Output> [IP] [SID] [PHID]

**Description**: informs NetSchedule that the job has been executed successfully. The command is becoming obsolete. Use PUT2 instead.

|  |  |  |
| --- | --- | --- |
| Parameter | Name | Description |
| JobKey | job\_key | String identifier: the job key. |
| RetCode | job\_return\_code | Integer: the execution return code |
| Output | output | String: the job output |
| IP | ip | IP address of a remote client  **Note**: Introduced in version 4.14.0. |
| SID | sid | Session ID of a remote client  **Note**: Introduced in version 4.14.0. |
| PHID | ncbi\_phid | CGI page hit ID  **Note**: Introduced in version 4.17.0 |

**Output synopsis**:

OK:

**Example**:

PUT JSID\_01\_6\_130.14.24.194\_9102 0 myoutput

OK:

### The PUT2 command

**Privileges**: any.

**Requires a queue**: yes

**NetSchedule output type**: single line

**Version**: 4.10.0

**Synopsis**:

PUT2 <JobKey> <SecurityToken> <RetCode> <Output> [IP] [SID] [PHID]

**Description**: informs NetSchedule that the job has been executed successfully. The client must be identified, i.e. the client\_node and client\_session parameters must be provided in the handshaking phase.

|  |  |  |
| --- | --- | --- |
| Parameter | Name | Description |
| JobKey | job\_key | String identifier: the job key. |
| SecurityToken | auth\_token | String identifier: the job security token which was provided in response to the GET2 command |
| RetCode | job\_return\_code | Integer: the execution return code |
| Output | output | String: the job output |
| IP | ip | IP address of a remote client  **Note**: Introduced in version 4.14.0. |
| SID | sid | Session ID of a remote client  **Note**: Introduced in version 4.14.0. |
| PHID | ncbi\_phid | CGI page hit ID  **Note**: Introduced in version 4.17.0 |

**Output synopsis**:

OK:

**Example**:

PUT2 JSID\_01\_6\_130.14.24.194\_9102 8764505\_3 0 myoutput

OK:

### The RETURN command

**Privileges**: any.

**Requires a queue**: yes

**NetSchedule output type**: single line

**Synopsis**:

RETURN <JobKey> [IP] [SID] [PHID]

**Description**: informs NetSchedule that the job has not been executed and should be moved to the pending state. The command is becoming obsolete. Use RETURN2 instead.

|  |  |  |
| --- | --- | --- |
| Parameter | Name | Description |
| JobKey | job\_key | String identifier: the job key. |
| IP | ip | IP address of a remote client  **Note**: Introduced in version 4.14.0. |
| SID | sid | Session ID of a remote client  **Note**: Introduced in version 4.14.0. |
| PHID | ncbi\_phid | CGI page hit ID  **Note**: Introduced in version 4.17.0 |

**Output synopsis**:

OK:

**Example**:

RETURN JSID\_01\_6\_130.14.24.194\_9102

OK:

### The RETURN2 command

**Privileges**: any.

**Requires a queue**: yes

**NetSchedule output type**: single line

**Version**: 4.10.0

**Synopsis**:

RETURN2 <JobKey> <SecurityToken> <Blacklist> [IP] [SID] [PHID]

**Description**: informs NetSchedule that the job has not been executed and should be moved to the pending state. The client must be identified, i.e. the client\_node and client\_session parameters must be provided in the handshaking phase.

|  |  |  |
| --- | --- | --- |
| Parameter | Name | Description |
| JobKey | job\_key | String identifier: the job key. |
| SecurityToken | auth\_token | String identifier: the job security token which was provided in response to the GET2 command |
| Blacklist | blacklist | Specifies whether the job is added to the worker node blacklist or not. Allowed values:  0 – do not add the job to the WN black list  1 – add the job to the WN blacklist (default)  **Note**: Introduced in version 4.17.0. |
| IP | ip | IP address of a remote client  **Note**: Introduced in version 4.14.0. |
| SID | sid | Session ID of a remote client  **Note**: Introduced in version 4.14.0. |
| PHID | ncbi\_phid | CGI page hit ID  **Note**: Introduced in version 4.17.0 |

**Output synopsis**:

OK:

**Example**:

RETURN2 JSID\_01\_6\_130.14.24.194\_9102 8764505\_3

OK:

### The RESCHEDULE command

**Privileges**: any.

**Requires a queue**: yes

**NetSchedule output type**: single line

**Version**: 4.19.0

**Synopsis**:

RESCHEDULE <JobKey> <SecurityToken> [Affinity] [Group] [IP] [SID] [PHID]

**Description**: informs NetSchedule that the job has not been executed and should be rescheduled (moved to the Pending state) with the provided affinity and group. The job will not be put to the worker node blacklist and the job run counter will be as it was before the job was given for execution. This command might be used to dispatch the job via changing its affinity and / or group. The client must be identified, i.e. the client\_node and client\_session parameters must be provided in the handshaking phase.

|  |  |  |
| --- | --- | --- |
| Parameter | Name | Description |
| JobKey | job\_key | String identifier: the job key. |
| SecurityToken | auth\_token | String identifier: the job security token which was provided in response to the GET2 command |
| Affinity | aff | Specifies the new job affinity. If not provided then the job will have no affinity.  **Note**: starting from NS 4.29.0 if no affinity is supplied then the server silently uses the ‘-‘ affinity. |
| Group | group | Specifies the new job group. If not provided then the job will have no group.  **Note**: starting from NS 4.29.0 if no group is supplied then the server silently uses the ‘-‘ group. |
| IP | ip | IP address of a remote client |
| SID | sid | Session ID of a remote client |
| PHID | ncbi\_phid | CGI page hit ID |

**Output synopsis**:

OK:

**Example**:

RESCHEDULE JSID\_01\_6\_130.14.24.194\_9102 8764505\_3 NewAffinity NewGroup

OK:

### The REDO command

**Privileges**: any.

**Requires a queue**: yes

**NetSchedule output type**: single line

**Version**: 4.28.0

**Synopsis**:

REDO <JobKey>

**Description**: instructs NetSchedule that the job should be moved to the Pending state. If a job is in Pending, Running or Reading state then the server reports an error and the command has no effect.

|  |  |  |
| --- | --- | --- |
| Parameter | Name | Description |
| JobKey | job\_key | String identifier: the job key. |

**Output synopsis**:

OK:

**Example**:

REDO JSID\_01\_6\_130.14.24.194\_9102 8764505\_3

OK:

### The WGET command

**Privileges**: any.

**Requires a queue**: yes

**NetSchedule output type**: single line

**Version**: it is obsolete starting from NetSchedule 4.10.0. Use GET2 instead.

**Synopsis**:

WGET <Port> <Timeout> [Aff] [IP] [SID] [PHID]

**Description**: provides a job for running. The command is becoming obsolete, use GET2 instead.

**Note:** starting from NS 4.17.0 a queue could be paused (see QPAUSE command). If the queue is paused then no job will be given if there are some jobs which match the requested criteria.

|  |  |  |
| --- | --- | --- |
| Parameter | Name | Description |
| Port | port | Integer: the port number on which the client will wait for a notification about vacant job availability if this WGET command did not pick a job. |
| Timeout | timeout | Integer: the timeout within which the client will wait for a notification about vacant job availability if this WGET command did not pick a job. |
| Aff | aff | A comma separated list of affinity identifiers.  If given then NetSchedule tries to pick a job with the mentioned affinities. This is the first priority of picking a job.  **Note**: starting from NS 4.29.0 the ‘-‘ affinity could be used here. It is a functional equivalent of a no affinity jobs. |
| IP | ip | IP address of a remote client  **Note**: Introduced in version 4.14.0. |
| SID | sid | Session ID of a remote client  **Note**: Introduced in version 4.14.0. |
| PHID | ncbi\_phid | CGI page hit ID  **Note**: Introduced in version 4.17.0 |

**Output synopsis (suitable job found)**:

OK:<JobKey> <“Input”> <“Aff”> <“ClientIP ClientSession”> <Mask>

**Output synopsis (no suitable job found)**:

OK:

|  |  |
| --- | --- |
| Parameter | Description |
| JobKey | String identifier: the job key. |
| Input | String: the job input. |
| Aff | String: the job affinity identifier. |
| ClientIP | String: the remote client IP. |
| ClientSession | String: the remote client session. |
| Mask | Integer: the job mask. |

**Example**:

WGET 10001 15

OK:JSID\_01\_6\_130.14.24.194\_9102 "input" "a2" "127.0.0.1 " 0

### The CWGET command

**Privileges**: any.

**Requires a queue**: yes

**NetSchedule output type**: single line

**Version**: 4.10.0

**Synopsis**:

CWGET [IP] [SID] [PHID]

**Description**: removes the client from the list of notifications. The client must be identified, i.e. client\_node and client\_session must be provided in the handshaking phase.

|  |  |  |
| --- | --- | --- |
| Parameter | Name | Description |
| IP | ip | IP address of a remote client  **Note**: Introduced in version 4.14.0. |
| SID | sid | Session ID of a remote client  **Note**: Introduced in version 4.14.0. |
| PHID | ncbi\_phid | CGI page hit ID  **Note**: Introduced in version 4.17.0 |

**Output synopsis**:

OK:

**Example**:

CWGET

OK:

### The FPUT command

**Privileges**: any.

**Requires a queue**: yes

**NetSchedule output type**: single line

**Synopsis**:

FPUT <JobKey> <ErrorMessage> <Output> <RetCode> [IP] [SID] [PHID]

**Description**: informs NetSchedule that the job execution has failed. The command is becoming obsolete. Use FPUT2 instead.

|  |  |  |
| --- | --- | --- |
| Parameter | Name | Description |
| JobKey | job\_key | String identifier: the job key. |
| ErrorMessage | err\_msg | String: the error message to associate with this execution try.  **Note**: NS 4.11.0 truncates error messages longer than 2048 bytes and adds MSG\_TRUNCATED at the end if so. |
| Output | output | String: the job output |
| RetCode | job\_return\_code | Integer: the execution return code |
| IP | ip | IP address of a remote client  **Note**: Introduced in version 4.14.0. |
| SID | sid | Session ID of a remote client  **Note**: Introduced in version 4.14.0. |
| PHID | ncbi\_phid | CGI page hit ID  **Note**: Introduced in version 4.17.0 |

**Output synopsis**:

OK:

**Example**:

FPUT JSID\_01\_6\_130.14.24.194\_9102 “DB access error” “” 2

OK:

### The FPUT2 command

**Privileges**: any.

**Requires a queue**: yes

**NetSchedule output type**: single line

**Version**: 4.10.0

**Synopsis**:

FPUT2 <JobKey> <SecurityToken> <ErrorMessage> <Output> <RetCode> [IP] [SID] [PHID] [NoRetries]

**Description**: informs NetSchedule that the job execution has failed. The client must be identified, i.e. the client\_node and client\_session parameters must be provided in the handshaking phase.

|  |  |  |
| --- | --- | --- |
| Parameter | Name | Description |
| JobKey | job\_key | String identifier: the job key. |
| SecurityToken | auth\_token | String identifier: the job security token which was provided in response to the GET2 command |
| ErrorMessage | err\_msg | String: error message to associate with this execution try.  **Note**: NS 4.11.0 truncates error messages longer than 2048 bytes and adds MSG\_TRUNCATED at the end if so. |
| Output | output | String: the job output |
| RetCode | job\_return\_code | Integer: the execution return code |
| IP | Ip | IP address of a remote client  **Note**: Introduced in version 4.14.0. |
| SID | sid | Session ID of a remote client  **Note**: Introduced in version 4.14.0. |
| PHID | ncbi\_phid | CGI page hit ID  **Note**: Introduced in version 4.17.0 |
| NoRetries | no\_retries | Instructs the server if the job should be failed regardless the retries setting. Acceptable values:  0 – consider the retries setting before making the decision of what state the job should be moved (default)  1 – move the job to the Failed state regardless of the retries setting.  **Note**: introduced in version 4.20.0 |

**Output synopsis**:

OK:

**Example**:

FPUT2 JSID\_01\_6\_130.14.24.194\_9102 8764505\_3 “DB error” myoutput 3

OK:

### The JXCG command

**Privileges**: any.

**Requires a queue**: yes

**NetSchedule output type**: single line

**Version**: some of the parameters have been introduced in NetSchedule 4.10.0, see the description below.

**Synopsis**:

JXCG <JobKey> <RetCode> <Output> [Aff] [IP] [SID] [PHID]

**Description**: exchange a job for another one. The command is becoming obsolete. Use PUT2 and GET2 instead.

**Note:** starting from NS 4.17.0 a queue could be paused (see QPAUSE command). If the queue is paused then no job will be given if there are some jobs which match the requested criteria.

|  |  |  |
| --- | --- | --- |
| Parameter | Name | Description |
| JobKey | job\_key | String identifier: the job key. |
| RetCode | job\_return\_code | Integer: the execution return code |
| Output | output | String: the job output |
| Aff | aff | A comma separated list of affinity identifiers.  If given then NetSchedule tries to pick a job with the mentioned affinities. This is the first priority of picking a job.  The allowed set of symbols for each identifier is [a-z][A-Z][0-9]\_  **Note**: starting from NS 4.29.0 the ‘-‘ affinity could be used here. It is a functional equivalent of no affinity jobs. |
| IP | ip | IP address of a remote client  **Note**: Introduced in version 4.14.0. |
| SID | sid | Session ID of a remote client  **Note**: Introduced in version 4.14.0. |
| PHID | ncbi\_phid | CGI page hit ID  **Note**: Introduced in version 4.17.0 |

**Output synopsis (suitable job found)**:

OK:<JobKey> <“Input”> <“Aff”> <“ClientIP ClientSession”> <Mask>

**Output synopsis (no suitable job found)**:

OK:

|  |  |
| --- | --- |
| Parameter | Description |
| JobKey | String identifier: the job key. |
| Input | String: the job input. |
| Aff | String: the job affinity identifier. |
| ClientIP | String: the remote client IP. |
| ClientSession | String: the remote client session. |
| Mask | Integer: the job mask. |

**Example**:

JXCG JSID\_03\_6\_130.14.24.194\_9102 0 myoutput aff=a1,a2

OK:JSID\_01\_6\_130.14.24.194\_9102 "input" "a2" "127.0.0.1 " 0

### The JDEX command

**Privileges**: any.

**Requires a queue**: yes

**NetSchedule output type**: single line

**Synopsis**:

JDEX <JobKey> <Timeout> [IP] [SID] [PHID]

**Description**: Delay the job expiration.

|  |  |  |
| --- | --- | --- |
| Parameter | Name | Description |
| JobKey | job\_key | String identifier: the job key. |
| Timeout | timeout | Integer: the timeout in seconds from now until the suggested new time when the run is considered as expired. If the current expiration exceeds the new one no changes will be made. |
| IP | ip | IP address of a remote client  **Note**: Introduced in version 4.14.0. |
| SID | sid | Session ID of a remote client  **Note**: Introduced in version 4.14.0. |
| PHID | ncbi\_phid | CGI page hit ID  **Note**: Introduced in version 4.17.0 |

**Output synopsis**:

OK:

**Example**:

JDEX JSID\_01\_3\_130.14.24.194\_9102 300

OK:

## Reader Commands

### The READ command

**Privileges**: any.

**Requires a queue**: yes

**NetSchedule output type**: single line

**Version**: 4.10.0 and up. The previous versions of the NetSchedule server had a different format. The old format is obsolete and is not described here.

**Synopsis**:

READ [Aff] [Port] [Timeout] [Group] [IP] [SID] [PHID] [AffinityMayChange] [GroupMayChange]

**Description**: Provides a job for reading. The client must be identified, i.e. the client\_node and client\_session parameters are provided in the handshaking phase.

**Note**: starting from NS 4.25.0 the command is affected by the current scope (see SETSCOPE). The scope restricts the candidate jobs. Both, empty scope identifier and the no-scope-only identifier are treated as the candidate job must not belong to any scope.

|  |  |  |
| --- | --- | --- |
| Parameter | Name | Description |
| Port | port | Integer: the port number on which the client will wait for a notification about vacant job availability if this READ command did not pick a job.  **Note**: Introduced in version 4.17.2 |
| Timeout | timeout | The timeout within which the client promises to finish reading the job.  Value 0 means that the queue configured timeout will be used.  **Note**: NS 4.17.2 treats the parameter as the timeout within which the reader will wait for a vacant job notification if there is no job for reading right now. |
| Aff | aff | A comma separated list of affinity identifiers.  If given then NetSchedule tries to pick a job with the mentioned affinities. This is the first priority of picking a job.  The allowed set of symbols for each identifier is [a-z][A-Z][0-9]\_  **Note**: implemented in NS 4.20.0 and up  **Note**: starting from NS 4.29.0 the ‘-‘ affinity could be used here. It is a functional equivalent of no affinity jobs. |
| Group | group | If given then only the jobs from the group will be provided.  **Note**: Starting from NetSchedule 4.23.0 a list of groups could be provided as a comma separated list.  **Note**: starting from NS 4.29.0 the ‘-‘ group could be used here. It is a functional equivalent of no group jobs. |
| AffinityMayChange | affinity\_may\_change | Integer: 0 or 1.  If 0 and an affinity restriction is provided then jobs in Pending and Running states will be affinity restricted when the no\_more\_jobs output parameter is calculated.  If 1 then the affinity restriction is not applied to the Pending and Running jobs because they could be rescheduled with changed affinity.  Default: 0  Introduced in NetSchedule 4.22.0 |
| GroupMayChange | group\_may\_change | Integer: 0 or 1.  If 0 and a group restriction is provided then jobs in Pending and Running states will be group restricted when the no\_more\_jobs output parameter is calculated.  If 1 then the group restriction is not applied to the Pending and Running jobs because they could be rescheduled with changed group.  Default: 0  Introduced in NetSchedule 4.22.0 |

**Output synopsis (the job for reading is found)**:

OK:job\_key=JobKey&auth\_token=SecurityToken&status=Status&client\_ip=IP&client\_sid=SID&ncbi\_phid=PHID&affinity=Affinity

**Output synopsis (the job for reading is not found)**:

OK:no\_more\_jobs=<bool>

|  |  |
| --- | --- |
| Parameter | Description |
| JobKey | The job key NetSchedule issued for the submitted job. |
| SecurityToken | The job security token which should be used for further read related operations. |
| Status | The job state from which the job was moved to the Reading state. The possible values here are Done and Failed (see the NetSchedule state diagram).  **Note**: NS 4.17.2 also allows Canceled here. |
| Affinity | The job affinity token  **Note**: introduced in NS 4.17.2 |
| no\_more\_jobs | Boolean. “true” if there are jobs which may become available later. “false” otherwise.  **Note 1**: introduced in NS 4.17.2 |
| IP | Client IP associated with the job at the submit time.  Introduced for NS 4.17.0 |
| SID | Client session ID associated with the job at the submit time.  Introduced for NS 4.17.0 |
| PHID | CGI page hit ID associated with the job at the submit time.  Introduced for NS 4.17.0 |

**Example**:

READ

OK:job\_key=JSID\_01\_5\_130.14.24.194\_9102&auth\_token=1461410271\_4&status=Done

### The READ2 command

**Privileges**: any.

**Requires a queue**: yes

**NetSchedule output type**: single line

**Version**: 4.20.0 and up

**Synopsis**:

READ2 <ReaderAff> <AnyAff> [ExclusiveNewAff] [Aff] [Port] [Timeout] [Group] [IP] [SID] [PHID] [AffinityMayChange] [GroupMayChange] [PrioritizedAff]

**Description**: Provides a job for reading. The client must be identified, i.e. the client\_node and client\_session parameters are provided in the handshaking phase.

**Note**: starting from NS 4.25.0 the command is affected by the current scope (see SETSCOPE). The scope restricts the candidate jobs. Both, empty scope identifier and the no-scope-only identifier are treated as the candidate job must not belong to any scope.

|  |  |  |
| --- | --- | --- |
| Parameter | Name | Description |
| Port | port | Integer: the port number on which the client will wait for a notification about vacant job availability if this READ2 command did not pick a job. |
| Timeout | timeout | Integer: the timeout within which the client will wait for a notification about vacant job availability if this READ2 command did not pick a job. |
| Aff | aff | A comma separated list of affinity identifiers.  If given then NetSchedule tries to pick a job with the mentioned affinities. This is the first priority of picking a job.  The allowed set of symbols for each identifier is [a-z][A-Z][0-9]\_  **Note**: starting from NS 4.29.0 the ‘-‘ affinity could be used here. It is a functional equivalent of no affinity jobs. |
| ReaderAff | reader\_aff | Integer: 0 or 1. If given then NetSchedule tries to pick a job bearing in mind the client preferred read affinities list. This is the second priority of picking a job. |
| AnyAff | any\_aff | Integer: 0 or 1. If 1 is given then any suitable job will be picked regardless of an affinity. This is the third priority of picking a job. |
| ExclusiveNewAff | exclusive\_new\_aff | Integer: 0 or 1. If set to 1 then NetSchedule tries to pick a job with no affinities or with an affinity which is not in the list of preferred read affinities of all readers i.e. exclusive new affinity. If the picked job has an affinity then it is added to the list of preferred read affinities of the reader.  This flag is allowed to set to 1 only if any\_aff is set to 0. |
| Group | group | If given then only the jobs from the group will be provided.  **Note**: Starting from NetSchedule 4.23.0 a list of groups could be provided as a comma separated list.  **Note**: starting from NS 4.29.0 the ‘-‘ group could be used here. It is a functional equivalent of no group jobs. |
| AffinityMayChange | affinity\_may\_change | Integer: 0 or 1.  If 0 and an affinity restriction is provided then jobs in Pending and Running states will be affinity restricted when the no\_more\_jobs output parameter is calculated.  If 1 then the affinity restriction is not applied to the Pending and Running jobs because they could be rescheduled with changed affinity.  Default: 0  Introduced in NetSchedule 4.22.0 |
| GroupMayChange | group\_may\_change | Integer: 0 or 1.  If 0 and a group restriction is provided then jobs in Pending and Running states will be group restricted when the no\_more\_jobs output parameter is calculated.  If 1 then the group restriction is not applied to the Pending and Running jobs because they could be rescheduled with changed group.  Default: 0  Introduced in NetSchedule 4.22.0 |
| PrioritizedAff | prioritized\_aff | Integer: 0 or 1. Could be set to 1 only if wnode\_aff, any\_aff and exclusive\_new\_aff are set to 0 and there is at least one affinity in the explicit affinity list.  If set to 1 then affinity in the explicit list are treated prioritized (first appeared has the highest priority) and a job is picked accordingly.  **Note**: introduced in version 4.22.0  **Note**: NS 4.27.0 and up let to have any\_aff set to 1 together with prioritized\_aff set to 1. If so then a job with any affinity (including no affinity) could be provided as the last resort. |

**Output synopsis (the job for reading is found)**:

OK:job\_key=JobKey&auth\_token=SecurityToken&status=Status&client\_ip=IP&client\_sid=SID&ncbi\_phid=PHID&affinity=Affinity

**Output synopsis (the job for reading is not found)**:

OK:no\_more\_jobs=<bool>

|  |  |
| --- | --- |
| Parameter | Description |
| JobKey | The job key NetSchedule issued for the submitted job. |
| SecurityToken | The job security token which should be used for further read related operations. |
| Status | The job state from which the job was moved to the Reading state. The possible values here are Done, Failed and Canceled (see the NetSchedule state diagram). |
| Affinity | The job affinity token |
| no\_more\_jobs | Boolean. “true” if there are no jobs which may become available for reading later. “false” otherwise. Even if “true” is returned, NetSchedule will send notifications if a new job matching provided criteria appears later. |
| IP | Client IP associated with the job at the submit time. |
| SID | Client session ID associated with the job at the submit time. |
| PHID | CGI page hit ID associated with the job at the submit time. |

**Example**:

READ2

OK:job\_key=JSID\_01\_5\_130.14.24.194\_9102&auth\_token=1461410271\_4&status=Done

### The CWREAD command

**Privileges**: any.

**Requires a queue**: yes

**NetSchedule output type**: single line

**Version**: 4.20.0

**Synopsis**:

CWREAD [IP] [SID] [PHID]

**Description**: removes the client from the list of read notifications. The client must be identified, i.e. client\_node and client\_session must be provided in the handshaking phase.

|  |  |  |
| --- | --- | --- |
| Parameter | Name | Description |
| IP | ip | IP address of a remote client |
| SID | sid | Session ID of a remote client |
| PHID | ncbi\_phid | CGI page hit ID |

**Output synopsis**:

OK:

**Example**:

CWREAD

OK:

### The SETRAFF command

**Privileges**: any.

**Requires a queue**: yes

**NetSchedule output type**: single line

**Version**: 4.20.0

**Synopsis**:

SETRAFF [AffToSet] [IP] [SID] [PHID]

**Description**: informs NetSchedule server about what the client list of preferred read affinities is. The client must be an identified one, i.e. the client\_node and client\_session parameters must be provided in the handshaking phase.

|  |  |  |
| --- | --- | --- |
| Parameter | Name | Description |
| AffToSet | aff | Comma or tab separated list of affinity identifiers to be set as the list of the client preferred affinities. The list can be empty.  The allowed set of symbols for each affinity identifier is [a-z][A-Z][0-9]\_  **Note**: starting from NS 4.29.0 if the ‘-‘ is supplied here then it is silently ignored. |
| IP | ip | IP address of a remote client |
| SID | sid | Session ID of a remote client |
| PHID | ncbi\_phid | CGI page hit ID |

**Output synopsis**:

OK:

**Example**:

SETRAFF a1,a2

OK:

### The CHRAFF command

**Privileges**: any.

**Requires a queue**: yes

**NetSchedule output type**: single line

**Version**: 4.20.0

**Synopsis**:

CHRAFF [AffToAdd] [AffToDel] [IP] [SID] [PHID]

**Description**: informs NetSchedule server about the changes in the client list of read preferred affinities. The client must be an identified one, i.e. the client\_node and client\_session parameters must be provided in the handshaking phase.

|  |  |  |
| --- | --- | --- |
| Parameter | Name | Description |
| AffToAdd | add | Comma or tab separated list of affinity identifiers to be added into the list of the client preferred affinities.  The allowed set of symbols for each affinity identifier is [a-z][A-Z][0-9]\_  **Note**: starting from NS 4.29.0 if the ‘-‘ is supplied here then it is silently ignored. |
| AffToDel | del | Comma or tab separated list of affinity identifiers to be removed from the list of the client preferred affinities.  The allowed set of symbols for each affinity identifier is [a-z][A-Z][0-9]\_  **Note**: starting from NS 4.29.0 if the ‘-‘ is supplied here then it is silently ignored. |
| IP | ip | IP address of a remote client |
| SID | sid | Session ID of a remote client |
| PHID | ncbi\_phid | CGI page hit ID |

**Output synopsis**:

OK:

**Example**:

CHRAFF a1,a2

OK:

### The CFRM command

**Privileges**: any.

**Requires a queue**: yes

**NetSchedule output type**: single line

**Version**: 4.10.0 and up. The previous versions of the NetSchedule server had a different format. The old format is obsolete and is not described here.

**Synopsis**:

CFRM <JobKey> <SecurityToken> [IP] [SID] [PHID]

**Description**: Confirms that the job has been read successfully. The client must be identified, i.e. the client\_node and client\_session parameters are provided in the handshaking phase.

|  |  |  |
| --- | --- | --- |
| Parameter | Name | Description |
| JobKey | job\_key | The job key. |
| SecurityToken | auth\_token | The job security token which was provided to the client in response to the READ command. |
| IP | ip | IP address of a remote client  **Note**: Introduced in version 4.14.0. |
| SID | sid | Session ID of a remote client  **Note**: Introduced in version 4.14.0. |
| PHID | ncbi\_phid | CGI page hit ID  **Note**: Introduced in version 4.17.0 |

**Output synopsis**:

OK:

**Example**:

CFRM JSID\_01\_5\_130.14.24.194\_9102 1461410271\_4

OK:

### The FRED command

**Privileges**: any.

**Requires a queue**: yes

**NetSchedule output type**: single line

**Version**: 4.10.0 and up. The previous versions of the NetSchedule server had a different format. The old format is obsolete and is not described here.

**Synopsis**:

FRED <JobKey> <SecurityToken> [ErrorMessage] [IP] [SID] [PHID] [NoRetries]

**Description**: Informs that reading of the job failed. The client must be identified, i.e. the client\_node and client\_session parameters are provided in the handshaking phase.

|  |  |  |
| --- | --- | --- |
| Parameter | Name | Description |
| JobKey | job\_key | The job key. |
| SecurityToken | auth\_token | The job security token which was provided to the client in response to the READ command. |
| ErrorMessage | err\_msg | The error message to associate with the failed reading. |
| IP | ip | IP address of a remote client  **Note**: Introduced in version 4.14.0. |
| SID | sid | Session ID of a remote client  **Note**: Introduced in version 4.14.0. |
| PHID | ncbi\_phid | CGI page hit ID  **Note**: Introduced in version 4.17.0 |
| NoRetries | no\_retries | Instructs the server if the job should be read failed regardless the read retries setting. Acceptable values:  0 – consider the retries setting before making the decision of what state the job should be moved (default)  1 – move the job to the ReadFailed state regardless of the read retries setting.  **Note**: introduced in version 4.20.0 |

**Output synopsis**:

OK:

**Example**:

FRED JSID\_01\_5\_130.14.24.194\_9102 1461410271\_4

OK:

### The RDRB command

**Privileges**: any.

**Requires a queue**: yes

**NetSchedule output type**: single line

**Version**: 4.10.0 and up. The previous versions of the NetSchedule server had a different format. The old format is obsolete and is not described here.

**Synopsis**:

RDRB <JobKey> <SecurityToken> [IP] [SID] [PHID] [Blacklist]

**Description**: Rollbacks the job reading. The client must be identified, i.e. the client\_node and client\_session parameters are provided in the handshaking phase.

|  |  |  |
| --- | --- | --- |
| Parameter | Name | Description |
| JobKey | job\_key | The job key. |
| SecurityToken | auth\_token | The job security token which was provided to the client in response to the READ command. |
| IP | ip | IP address of a remote client  **Note**: Introduced in version 4.14.0. |
| SID | sid | Session ID of a remote client  **Note**: Introduced in version 4.14.0. |
| PHID | ncbi\_phid | CGI page hit ID  **Note**: Introduced in version 4.17.0 |
| Blacklist | blacklist | Specifies whether the job is added to the reader blacklist or not. Allowed values:  0 – do not add the job to the reader black list  1 – add the job to the reader blacklist (default)  **Note**: Introduced in version 4.22.0. |

**Output synopsis**:

OK:

**Example**:

RDRB JSID\_01\_5\_130.14.24.194\_9102 1461410271\_4

OK:

### The REREAD command

**Privileges**: any.

**Requires a queue**: yes

**NetSchedule output type**: single line

**Version**: 4.28.0

**Synopsis**:

REREAD <JobKey>

**Description**: instructs NetSchedule that the job should be moved to the state the job was before it was given for reading. If a job is in Pending, Running or Reading state then the server reports an error and the command has no effect. If a job has not been read yet then the server reports a warning and the command has no effect.

|  |  |  |
| --- | --- | --- |
| Parameter | Name | Description |
| JobKey | job\_key | String identifier: the job key. |

**Output synopsis**:

OK:

**Example**:

REREAD JSID\_01\_6\_130.14.24.194\_9102 8764505\_3

OK:

## Obsolete Commands

The following commands:

* REGC
* URGC
* INIT

Are obsolete and the output for them will always be:

OK:WARNING:Obsolete;

The following commands:

* JRTO
* FRES
* QERY
* QSEL
* MONI
* LOG
* JDEX2
* STSN

Are obsolete and the output for them will always be:

ERR:Not implemented

The command:

* STAT WNODE

Is obsolete and the output for it will always be:

OK:WARNING:Obsolete, use STAT CLIENTS instead;

# Notification Formats

NetSchedule supports three types of notifications:

* Notifications which are sent when a job changes its state. These notifications are sent when a job is submitted with non-zero parameters port and timeout.
* Notifications which are sent when a job for execution becomes available for worker nodes which issued the WGET or GET2 commands with a port and a timeout and there were no jobs available for them.
* Notifications which are sent when a job for reading becomes available for readers which issued the READ command with a port and a timeout and there were no jobs available for them. These notifications are very similar to the GET2 notifications. In fact the difference is only the ‘reason’ field value in a notification packet, see below.

## Job State Changed Notification Format

**Version**: 4.10.0 and up. The older version had another format and it is not supported any more.

ns\_node=<NetScheduleNodeID>&job\_key=<JobKey>&job\_status=<State>&last\_event\_index=<Index>&reason=<Reason>&msg=<ProgressMessage>&msg\_truncated=<TruncatedCount>

|  |  |  |
| --- | --- | --- |
| Parameter | Name | Description |
| ns\_node | NetScheduleNodeID | String identifier.  NetSchedule node identifier. The identifier stays the same between NetSchedule restarts if so. |
| job\_key | JobKey | String identifier.  The job key which changed its state. |
| job\_status | State | String identifier.  The new job state as a string, e.g. Done. |
| last\_event\_index | Index | Integer.  0-based index of the last event recorded for a job. |
| reason | Reason | The reason why the notification is sent. The possible values are:   * “status”. The job status has been changed. * “stolen”. The job notification host:port has been changed. This notification is sent only if the client requested it. See the description in the LISTEN command. * “progress”. The job progress message has been changed. This notification is sent only if the client requested it. See the description in the LISTEN and SUBMIT commands.   **Note:** introduced in version 4.31.0 |
| msg | ProgressMessage | The URL encoded job progress message. The value may have up to 768 characters. If there is more then the message is truncated and the number of the truncated characters is supplied in the msg\_truncated parameter.  **Note:** introduced in version 4.31.0 |
| msg\_truncated | TruncatedCount | The number of characters truncated in the original job progress message to meet the URL encoded length criteria condition. If there were no truncation then the parameter does not appear in the notification.  **Note:** introduced in version 4.31.0 |

## Job Available Notification Format

The format differs depending what command was used by a worker node. The old worker nodes use the WGET command and the provided information is limited to what they can accept. The newer worker nodes use GET2 command and the format includes additional information.

NS 4.17.2 and up also supports READ notifications, see below.

WGET format:

NCBI\_JSQ\_<QueueName>

|  |  |  |
| --- | --- | --- |
| Parameter | Name | Description |
| NCBI\_JSQ\_ |  | Fixed string |
| QueueName | QueueName | Queue name which has a job available for a worker node. |

GET2 format (NS 4.17.1 and below):

ns\_node=<NetScheduleNodeID>&queue=<QueueName>

GET2 format (NS 4.17.2 and up):

reason=get&ns\_node=<NetScheduleNodeID>&queue=<QueueName>

READ format (NS 4.17.2 and up):

reason=read&ns\_node=<NetScheduleNodeID>&queue=<QueueName>

|  |  |  |
| --- | --- | --- |
| Parameter | Name | Description |
| ns\_node | NetScheduleNodeID | String identifier.  NetSchedule node identifier. The identifier stays the same between NetSchedule restarts if so. |
| QueueName | QueueName | Queue name which has a job available for a worker node. |