Xi Software

GNUbatch Release 1

Web Browser Interface

This manual is for GNUbatch (Web Browser Interface Manual).

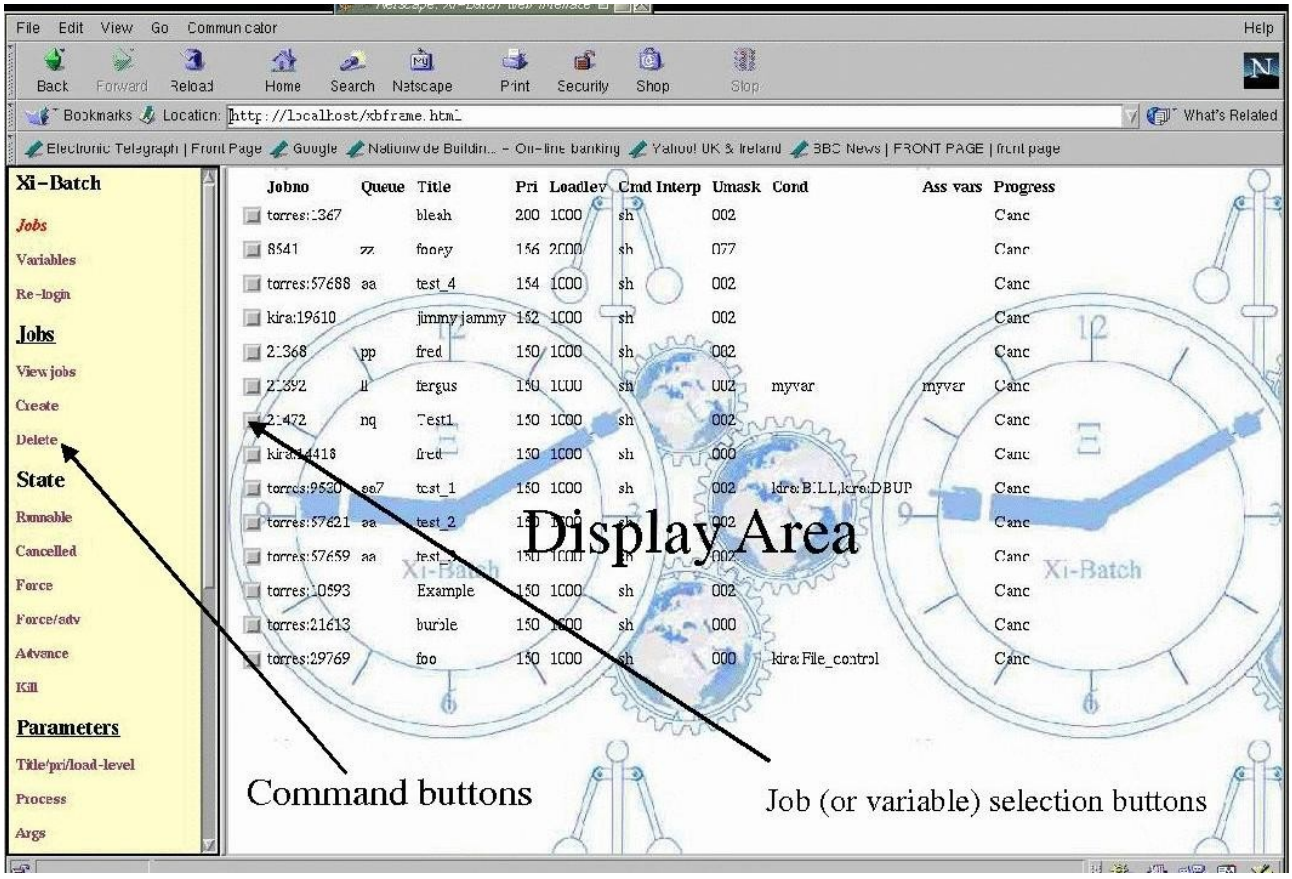
Table of Contents

1 GNUbatch Web Browser

1.1 Overview

The **GNUbatch** web browser provides most of the facilities of the queue viewers gbch-q, gbch-xq and gbch-xmq via a web-browser interface. Local and remote jobs and variables may be accessed, attributes changed, jobs created, started and halted.

The display is similar to the following:



The interface consists of a series of HTML files, a JavaScript function library and a series of CGI binaries. The background display, colours, styles and some of the functionality may all be easily adjusted as required.

Most of the job or variable operations may be effected by clicking one or more of the checkboxes displayed beside the relevant job or variable and the desired operation on the command list on the left.

Prior to access, the user is invited to log in with his or her username and password, or a default user may be specified for the interface. The user may "re-log-in" with a different user name and password to access facilities available to the new user if required.

1.1.1 Varieties of interface

There are two versions of the browser interface:

1. A local version, for where the web server and **GNUbatch** are running on the same host. (The version of **GNUbatch** does not have to be networked).
2. A remote version, for where the web server and **GNUbatch** are running on different hosts (in fact this is implemented as API programs).

The interface is virtually identical in each case, except that the initial login screen requests a host name in addition to the user name and password and some errors, particularly "no permission" errors, are detected slightly differently.

1.2 Login

The first time the **GNUbatch** web interface is accessed, the following is usually displayed:

Log in with your usual Unix user name and password[1] and press the "Login as User" button.

If you are using the remote interface (supporting **GNUbatch** on a different host from the web server) the login screen will look like this:

In this case you should specify the host name on which **GNUbatch** is being run as well (you can put `localhost` if it is in fact the same host as the server).

It may be that the interface is set up with a default user name (and possibly host name), in which case this step will be omitted, but the permissions on the default user name are likely to be limited.

1 The system may be set up to use a different set of passwords from the usual login passwords.

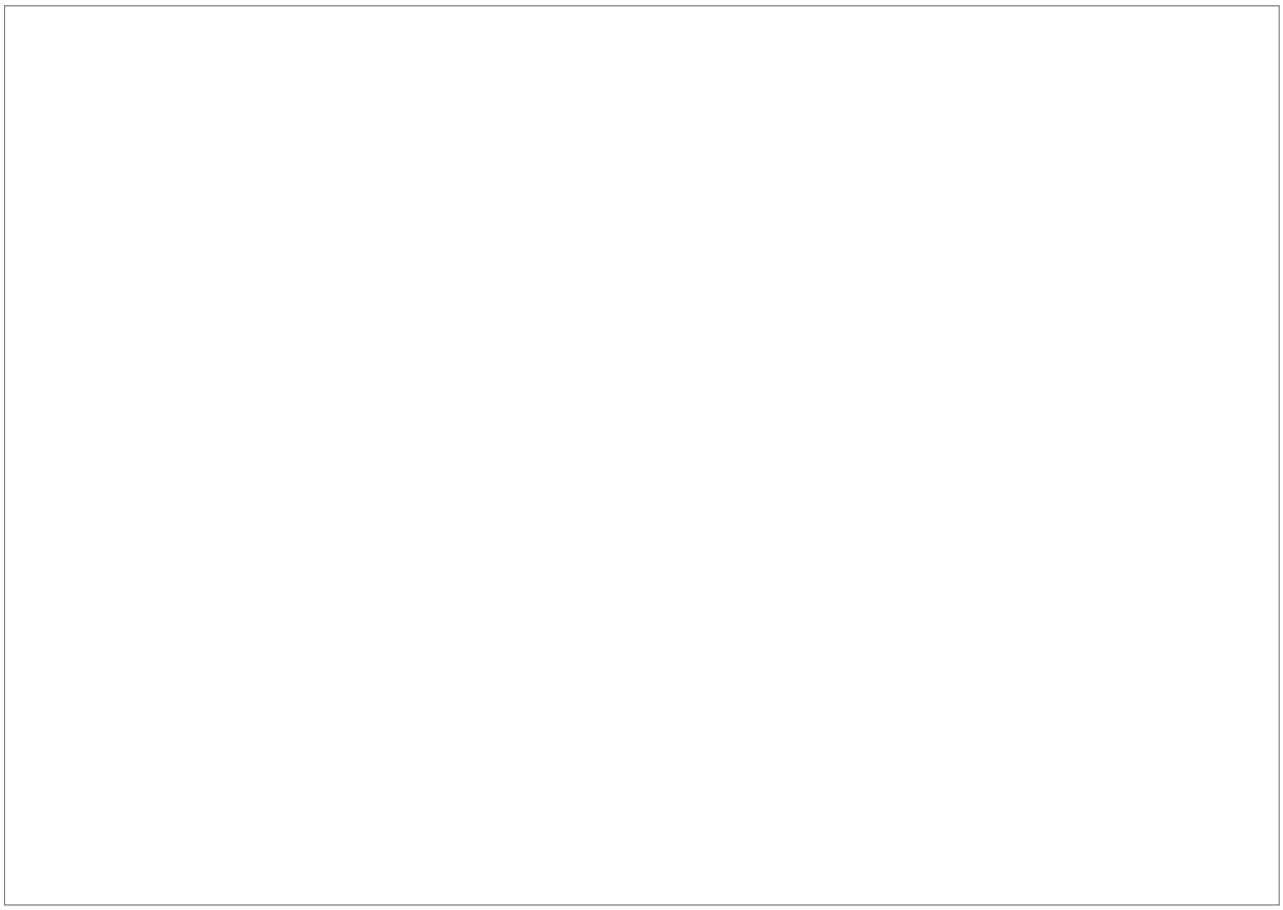
At any stage you may select a new user name (and/or host name) by pressing the **Re-login** link on the button bar. Note that a "cookie" is saved by your browser to record a code which is passed with each request to identify you. It is important to enable cookies on your browser for this reason. Certain options are also saved as cookies also.

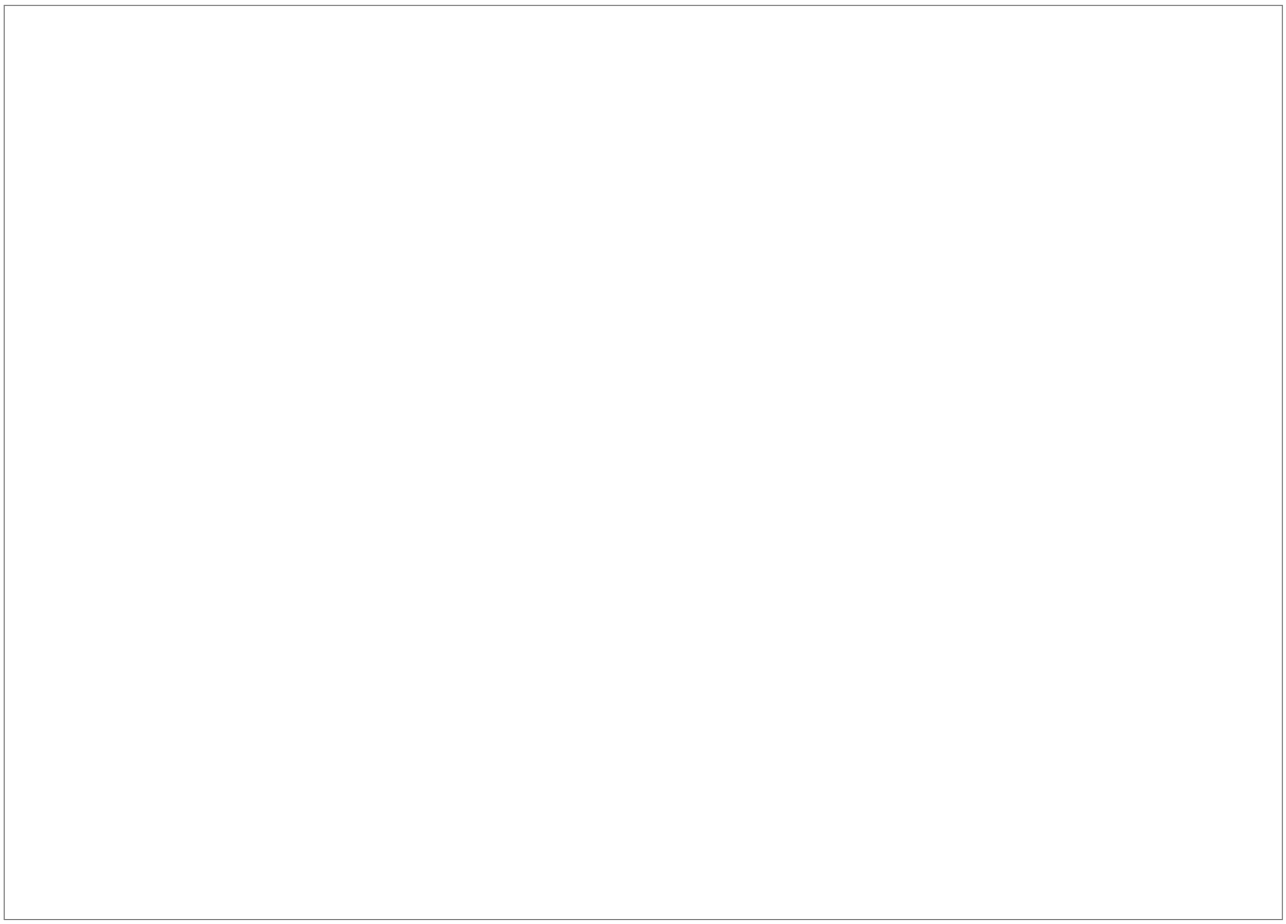
After a certain time, which may be configured by the administrator as so many days or hours, the login record will time out and become stale. You will then have to re-login.

After a successful login, the main display of jobs will appear on the right half of the screen.

1.3 Switching between job and variable lists

After successfully logging-in, the initial display will be of the job list. The top two entries in the button bar on the left allow the user to toggle between the display of jobs and variables. Note that the button bar is updated as well as the display.





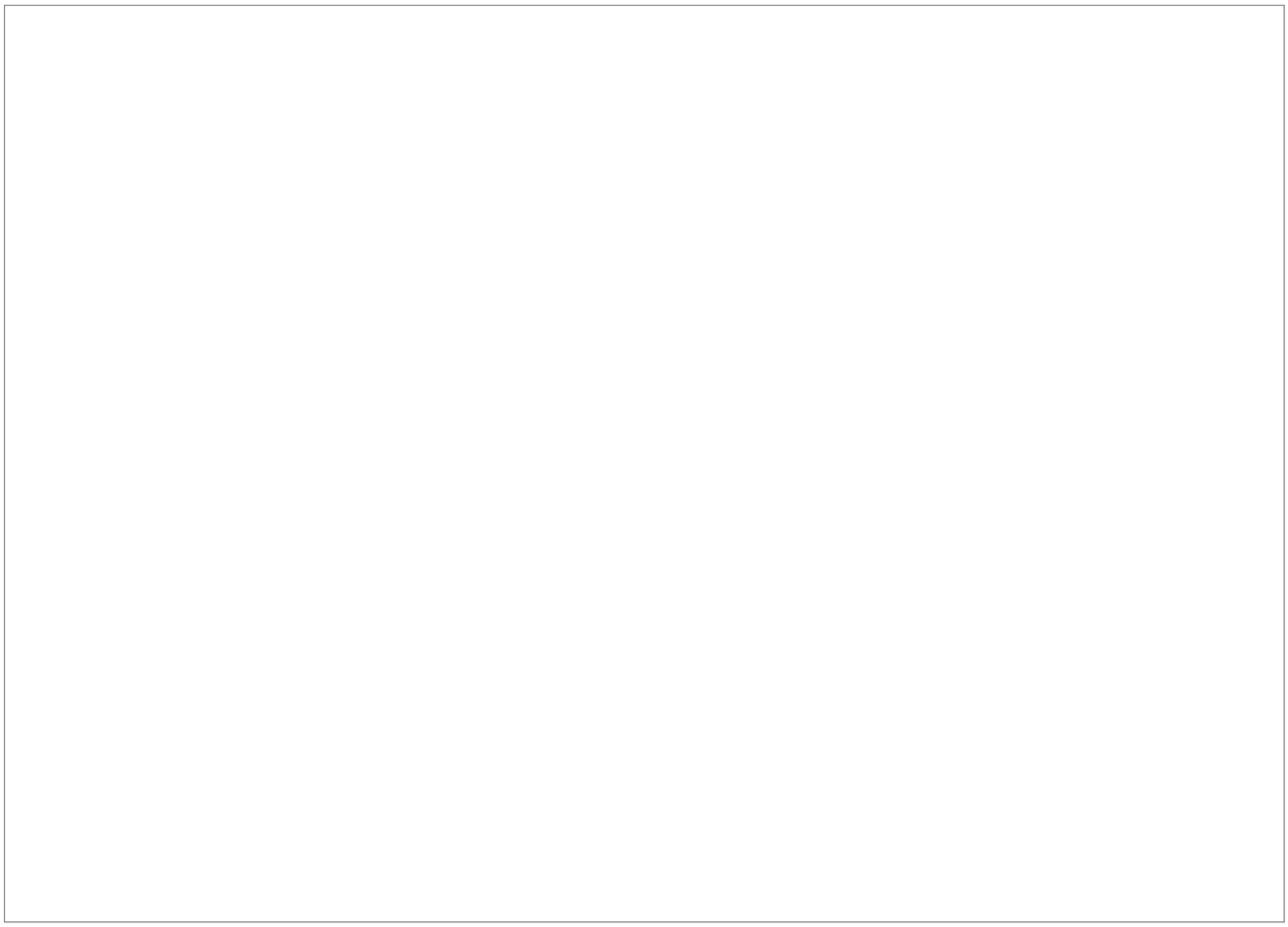
Nothing in either display implies that the user has any right to access anything in the displayed list. Appropriate error messages will be displayed if the user attempts access to facilities which he or she does not have permission.

1.4 List formats

The displayed attributes of the job or variable lists may be adjusted to suit the user's preferences. The selected formats are saved as a "cookie" in the web browser and will be re-selected each time the user returns to the display.

To change the format of the displayed list (either job or variable), click on the **Set Format** button on the button bar.

The main display will change to a screen of the form:

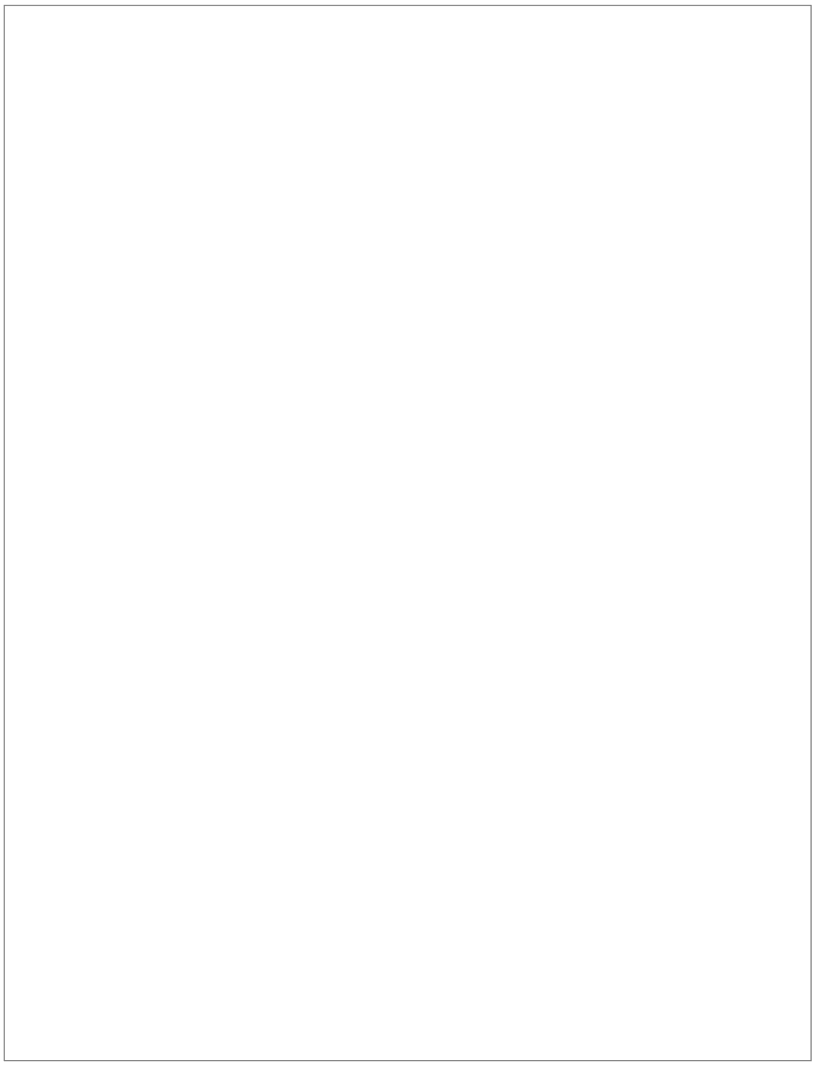


Each row corresponds to a column of the job or variable list (according to whichever was being displayed at the time).

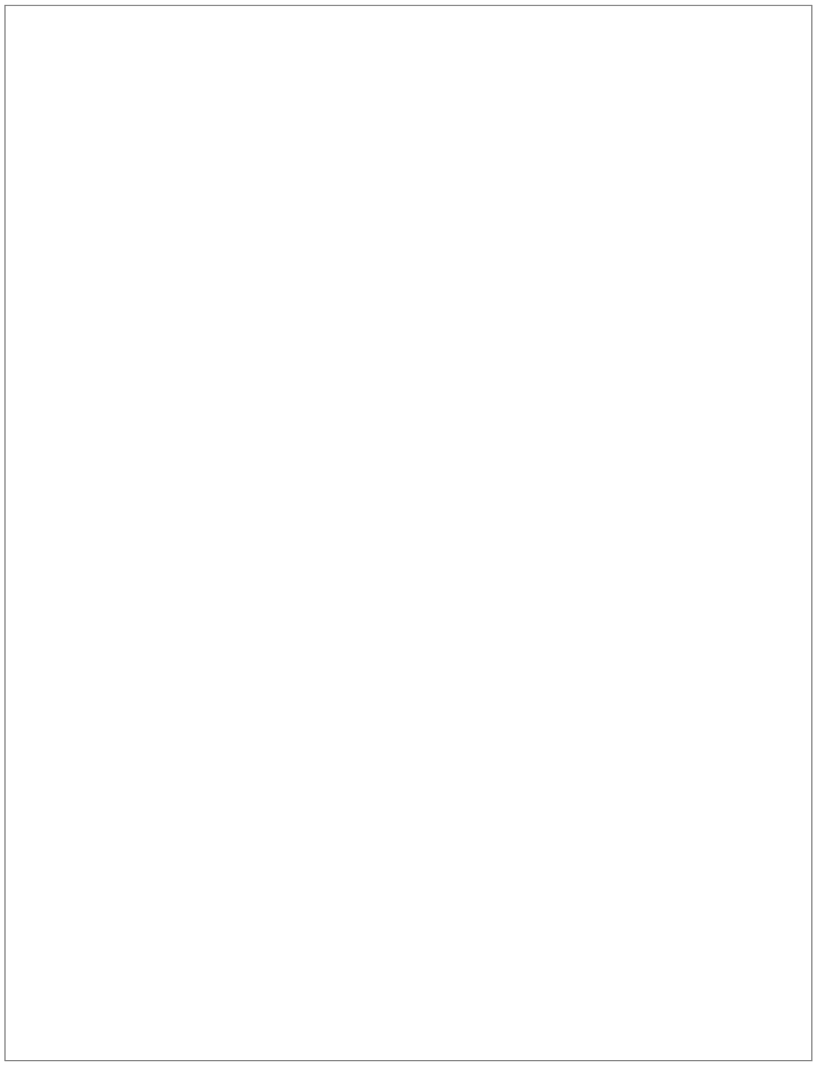
To build up the required format, start with the attribute to be displayed in the leftmost column of the result, and work across to the right. To select a field, decide whether you wish the field to be displayed left or right-aligned or centred (the individual fields have common defaults set), and click the button on the left to insert the field.

This will cause a "pop-up" window to be displayed as follows:

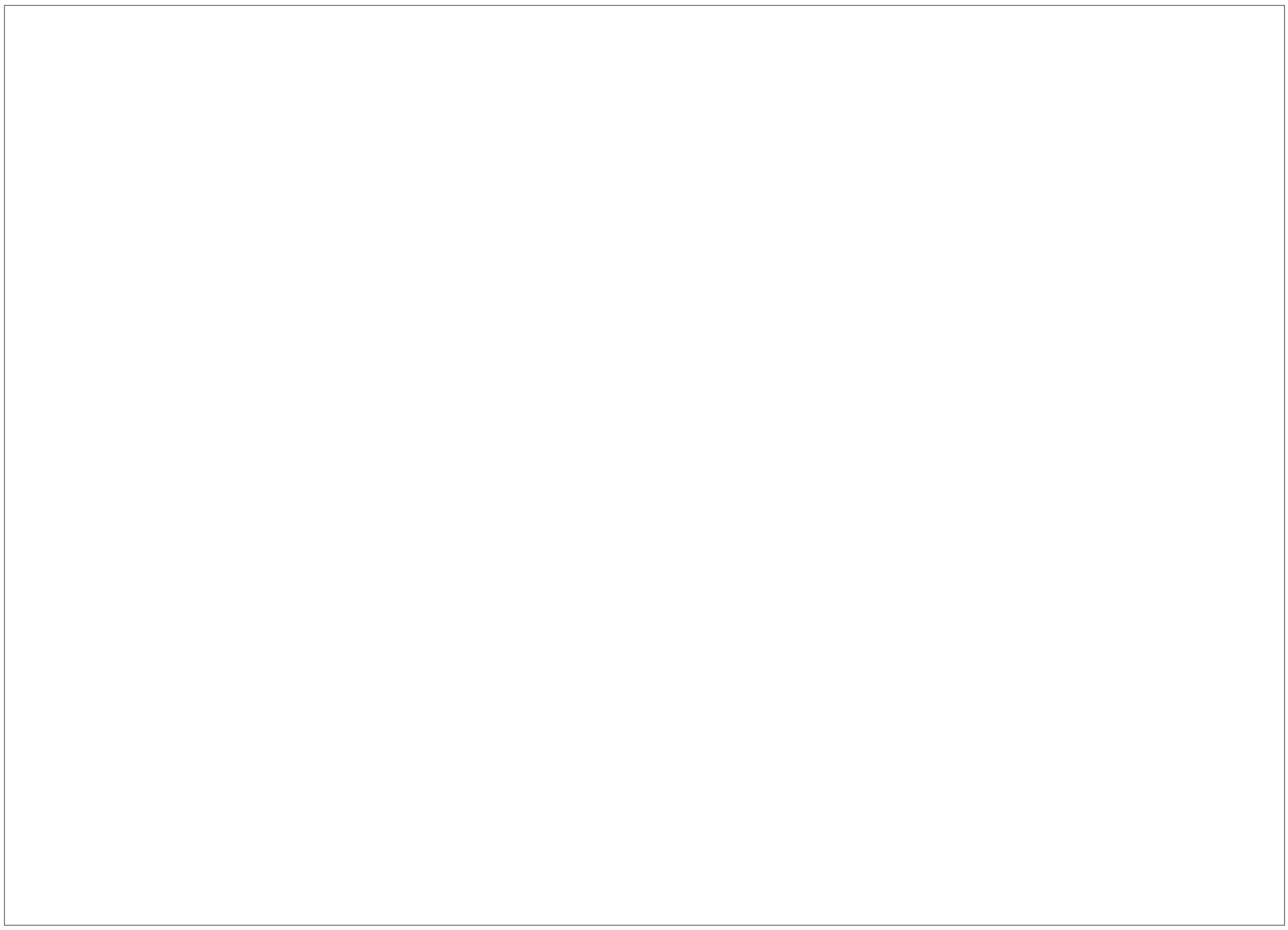
The "code" is for reference and is used internally. The other two fields give the alignment selected and the description of the field.



As you continue to work through the fields required, the result window will be updated, for example:



When you have finished doing this, press **Set Formats** to change the format display, in this instance to



You can also press **Cancel** to abort the operation, or **Reset Default** to revert to the standard format and reset the "cookie".

The standard format may be defined as required for each installation by editing the parameters file.

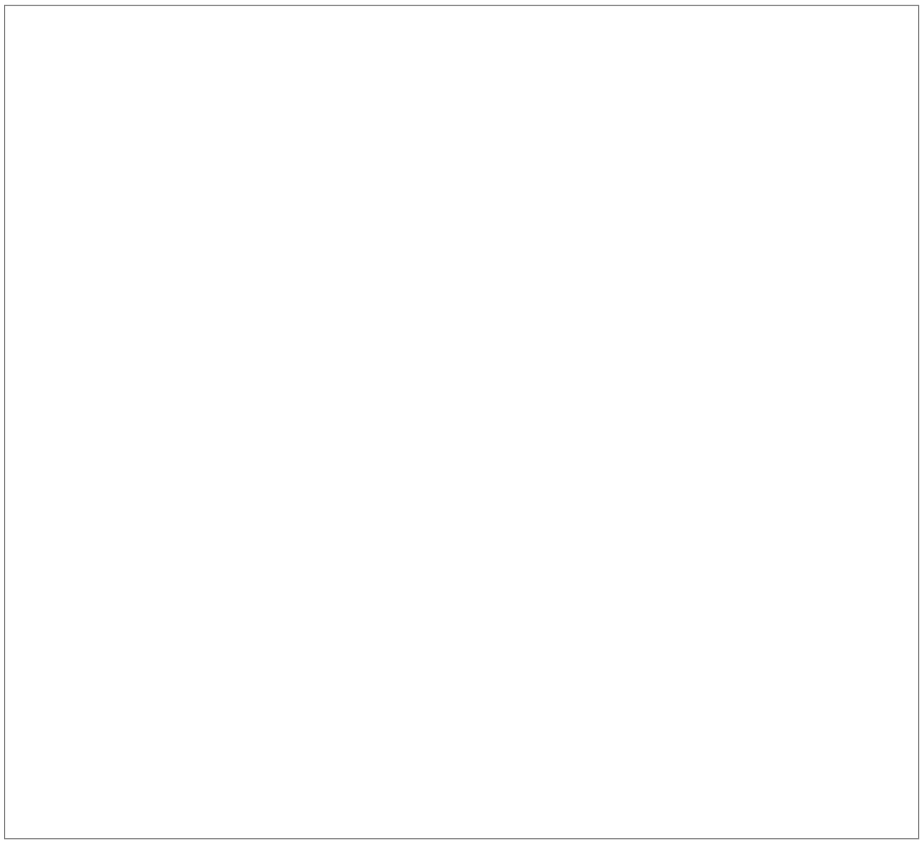
1.5 Display option setting

You can arrange control which jobs or variables are displayed by clicking on the Options button. The selected options are saved as a "cookie" and retained for future invocations of the job or variable list display.

1.5.1 Job Options

If jobs are being displayed, the following form will appear:

The initial selections are the default, and are the most inclusive. To revert to the default format, just click on **Set Options** without altering anything.



1.5.1.1 Display column headers

Check or uncheck this to determine whether the first row of the job display is preceded by column titles.

1.5.1.2 Sort jobs by time order

Check this to sort the jobs in the order they are set to execute, rather than by priority on the queue.

1.5.1.3 Display job queue prefix

You can put a queue name in here to cause the display to be limited to jobs with that prefix. You can also put a comma-separated list of queue names, and you can use a shell-style "wildcard", for example: ac*,*[1-6] would select jobs with queue names starting with ac or ending with 1 to 6.

1.5.1.4 Include null job queue prefix

Is applicable only if a queue prefix name is given. In this case it determines whether or not a job is displayed if it has no queue prefix.

1.5.1.5 Limit to user

Limit the display to jobs owned by the given user, or users separated by commas, or using wild cards as for job queues.

1.5.1.6 Limit to group

Limit the display to jobs owned by the given group, or groups separated by commas, or using wild cards as for job queues.

1.5.1.7 Limit display of jobs

This selects whether all jobs are displayed, just jobs local to the server, or ones local to the server plus jobs which could be remote-executed by the server.

1.5.2 Variable Options

Again the initial selections are the default.

1.5.2.1 Display column headers

Selects or deselects the title at the top of each column.

1.5.2.2 Limit to user/limit to group

This limits the display of variables to the user or group, or set of users and groups as with jobs.

1.5.2.3 Display local variables only

If set, only variables local to the server (or selected server in the case of the remote interface) are displayed.

2 Job Actions

The following take actions without changing parameters of the job or jobs.

Unless otherwise stated, the operations may be performed upon several jobs at once. In all cases the jobs should be selected by clicking in the box on the left of the display row.

Note that you can clear all the boxes on the job or variable list by pressing **Clear Marked**.

The following operations are documented here:

1. View Jobs
2. Delete Jobs
3. Set runnable
4. Set cancelled
5. Force
6. Force and advance
7. Advance
8. Kill

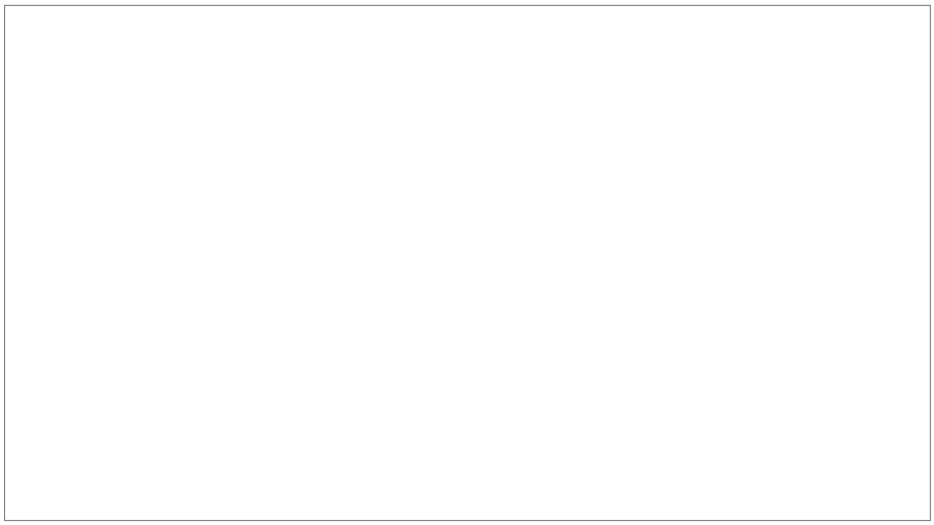
2.1 View jobs

This causes a window to be opened for each selected job to display the script for the job, as follows:

The window is scrollable and resizeable. When finished, click on **Quit**, or close the window using the control.

2.2 Delete Jobs

This causes the specified jobs to be deleted. If all goes well, you will receive a completion message, thus:



2.3 Set runnable/set cancelled

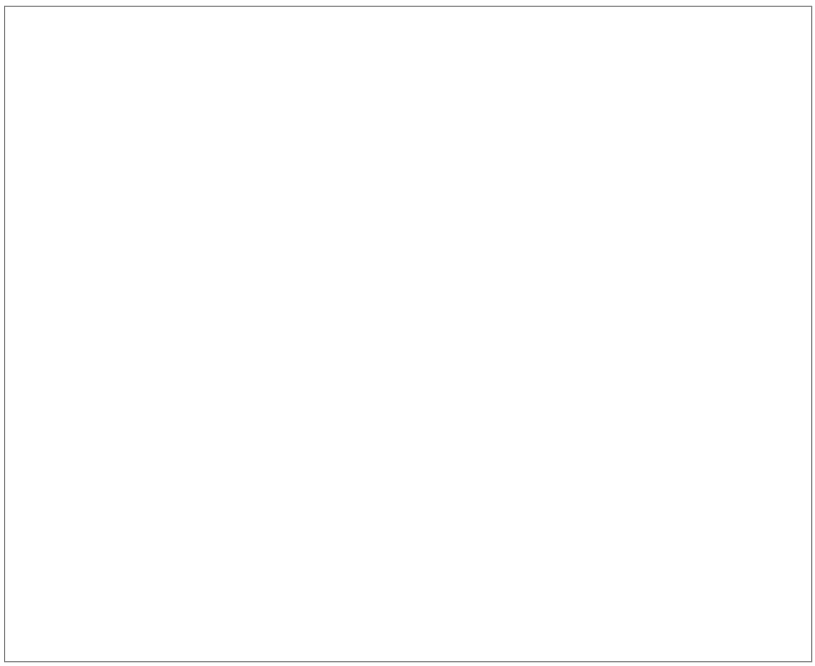
These cause the selected jobs to be set to the selected state. If all goes well, you will receive a completion messages as per deletion.

2.4 Force, Force and advance, Advance

These apply the specified operation to the selected jobs. If all goes well, you will receive a completion messages as per deletion.

2.5 Kill jobs

This gives a window as follows, requesting the desired signal type to be applied to the jobs:



If all goes well, you will receive a completion messages as per deletion.

3 Job Parameters

The following operations are available to change various job parameters.

In all cases, only one job at a time should be selected, by clicking on the box at the extreme left of the job row.

Note that you can clear all the boxes on the job or variable list by pressing **Clear Marked**.

The following operations are documented here.

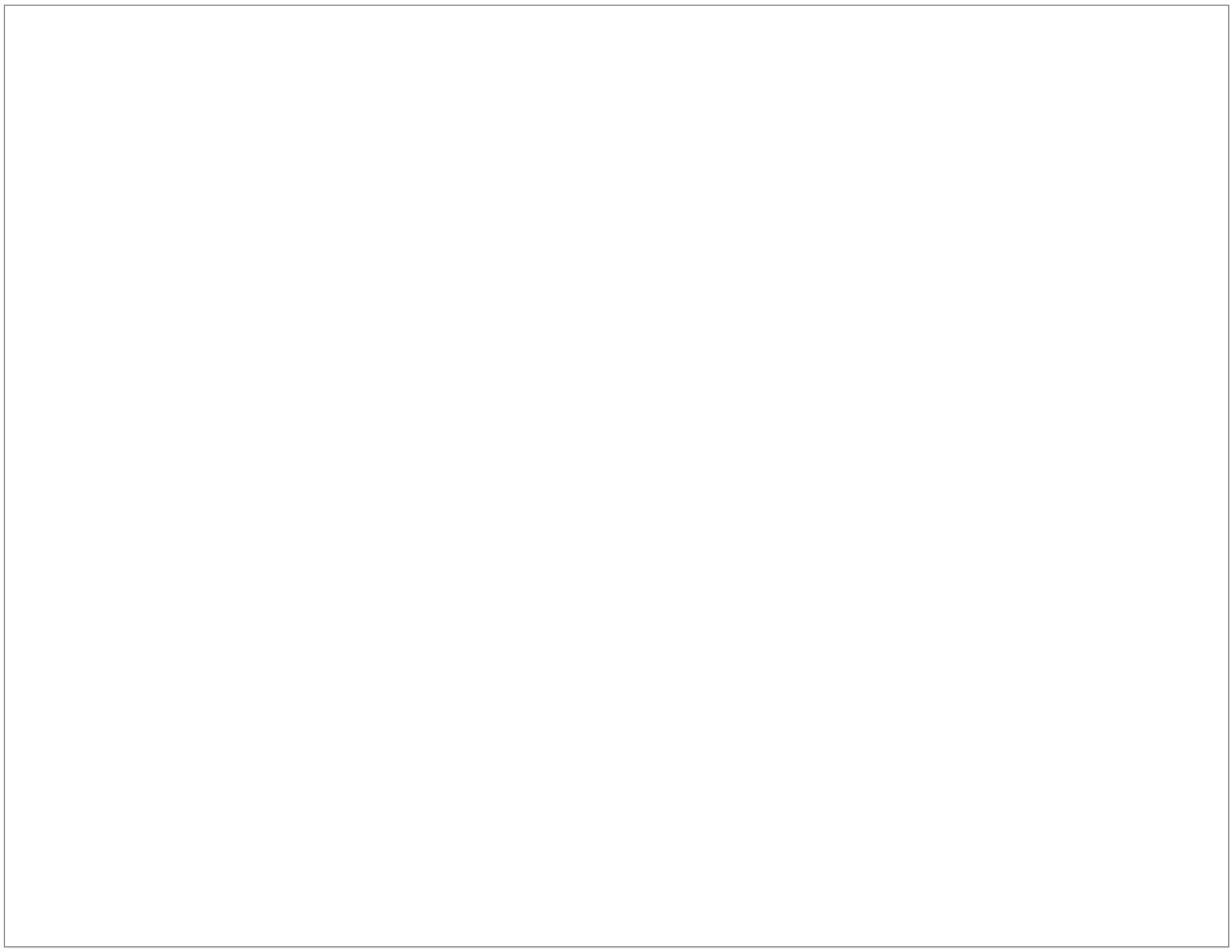
1. Create Job
2. Create Job from file
3. Set title, priority, command interpreter, load level
4. Set process parameters
5. Set mail/write flags
6. Set permissions
7. Set arguments
8. Set I/O redirections
9. Set time
10. Set conditions
11. Set assignments

3.1 Create Job

On the browser interface, jobs are created by setting up a skeleton job and adding or changing parameters.

Click on the **Create** button to give a window of the form:

The fields here have the following functions:



3.1.1 Queue

You can enter a queue prefix here, or you can select an existing queue name from the selection box on the right, which will insert the queue name.

3.1.2 Title

Enter any title string here.

3.1.3 Directory

Enter the working directory here. It is initialised to the home directory for the logged-in user.

3.1.4 Command interpreter

As with queue name, this can be inserted by selecting from the selection box on the right.

3.1.5 Create job in cancelled state/create ready to run

Usually you will want to add options later, so select the cancelled state unless a very

simple job is intended.

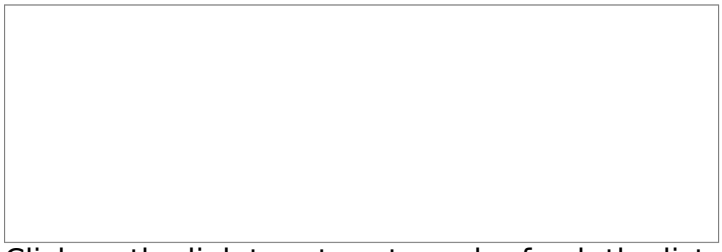
3.1.6 Add newline to end

People keep forgetting to do this in the box below, so this remembers for you.

3.1.7 Script body

Enter the script here.

When the job is submitted, you will receive a message as follows:



Click on the link to return to and refresh the list of jobs.

3.2 Create job from file

This is almost identical to the Create Job option, except that instead of editing a text area, the job is created from a file on the client system whose name is typed in, or selected using the Browse button displayed.

3.3 Title, Priority, Command Interpreter, Load Level

To set or reset these, select one job, and click on the button on the button bar, to give the following window.

As with creating jobs, as well as typing in a queue name or command interpreter name, you can use the selection box on the right to select one of the pre-existing

values.

Priority is set using the two selection boxes to give tens (up to 250) and units. Load level digits may likewise be set as thousands to units on the selection boxes.

The fields are all initialised with the current values for the job.

Press **Make Changes** to continue, giving a confirmation message, thus

or **Cancel** to return to the job list. **Reset** returns the field to their initial states.

3.4 Process parameters

To set or reset these, select a job and press the button to get the following window:

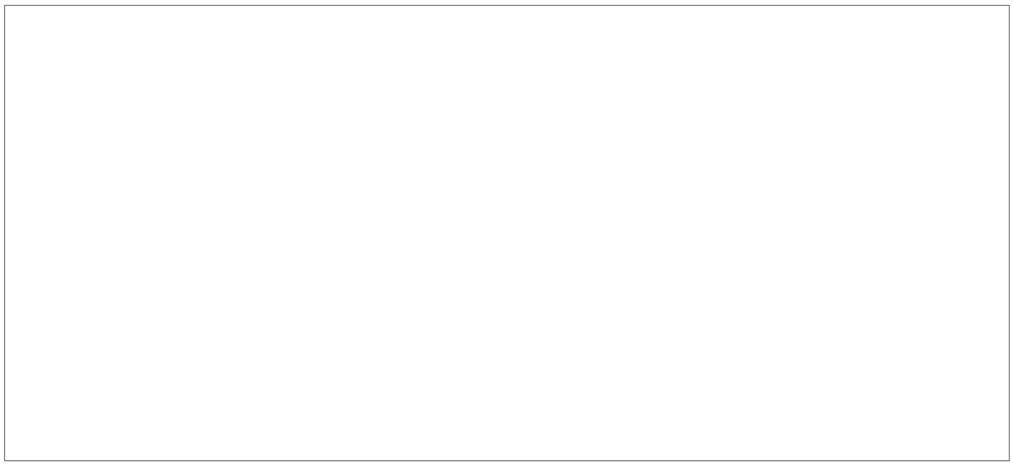
The fields are as documented in the System Reference Manual under process parameters.

If all goes well, you will receive a confirmation message, as above.

3.5 Set mail/write flags

To set the mail or send message on completion flags for a job or jobs, select them and click the **Mail/Write** button.

The following window appears:



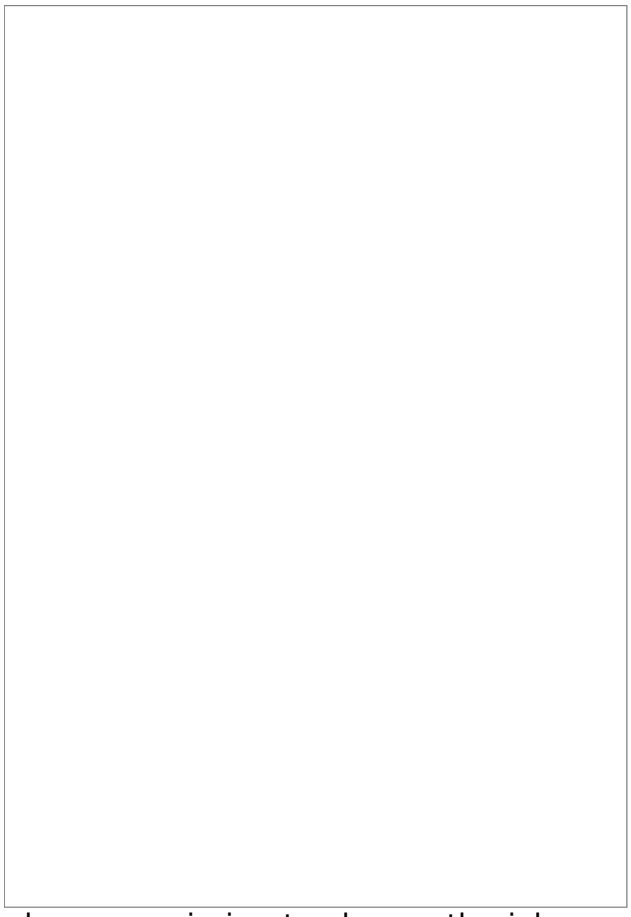
Click on the relevant check box. These default to being off and only what you change is applied to the job, so if you want to turn the flags off, be sure to click them once as

if to set them and then again to unset them before pressing **Set Flags**.

If all goes well, you will receive a confirmation message, as above.

3.6 Permissions

To set the permissions for a job, click on it to have the existing permissions displayed as follows:



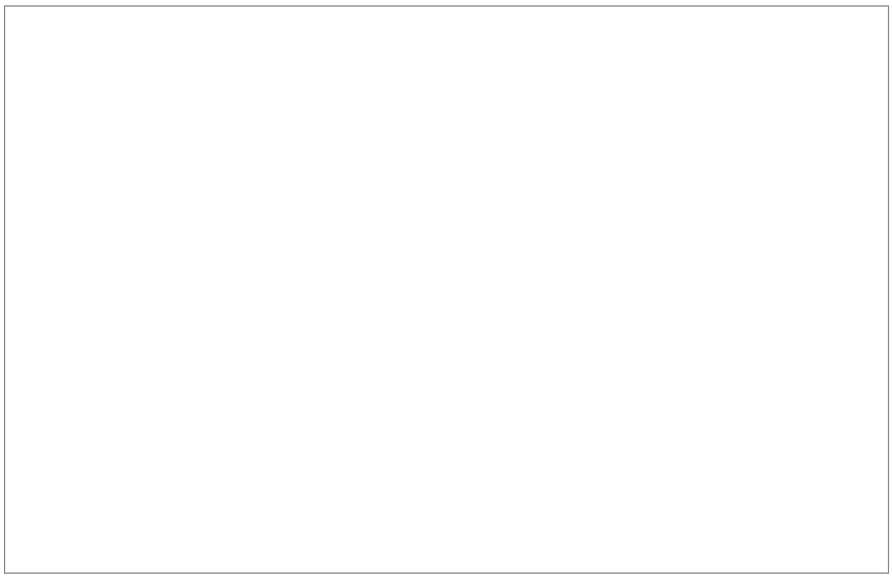
If you currently have permission to change the job, you can set or clear appropriate permission boxes and press **Make Changes** to apply the new permissions.

If all goes well, you will receive a confirmation message, as above.

3.7 Arguments for job

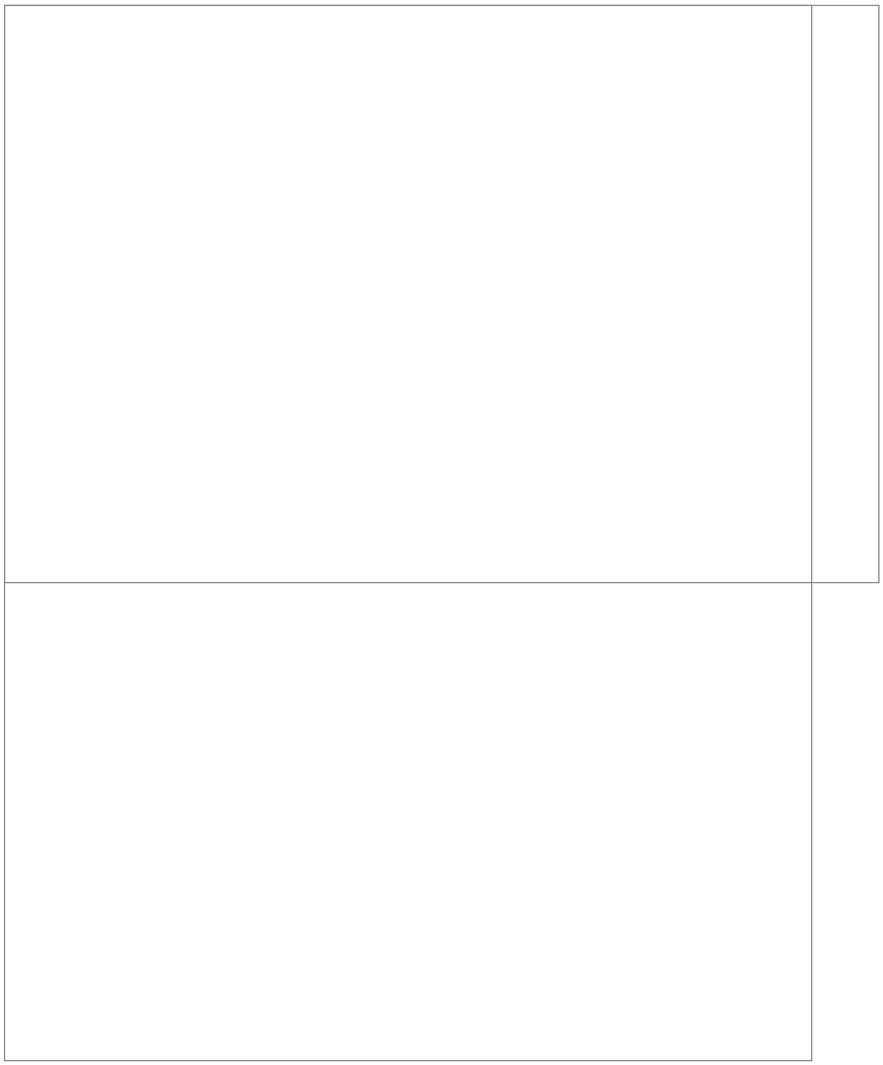
To set arguments to be passed to the job, select the job and click on the **Args** button. If the job has no arguments, the following window will be displayed:

or if there are existing arguments, the window will look like this:



The way in which this operates is that you build up a new list of arguments (which is displayed in additional window) by clicking on the **Add to new** button corresponding to the argument to be added to a new list (which you can edit first). So if you wanted to change the arguments to upper case and add an extra one, you could change the boxes to read:

Then click on each **Add to new** button in turn so as to get a new window thus:



Then click on **Make Changes** on the main screen to apply the new list of arguments built up, or **Cancel** to abandon it.

You can click on the **Add to New** buttons in any order to include, duplicate, change, add or omit arguments in any order.

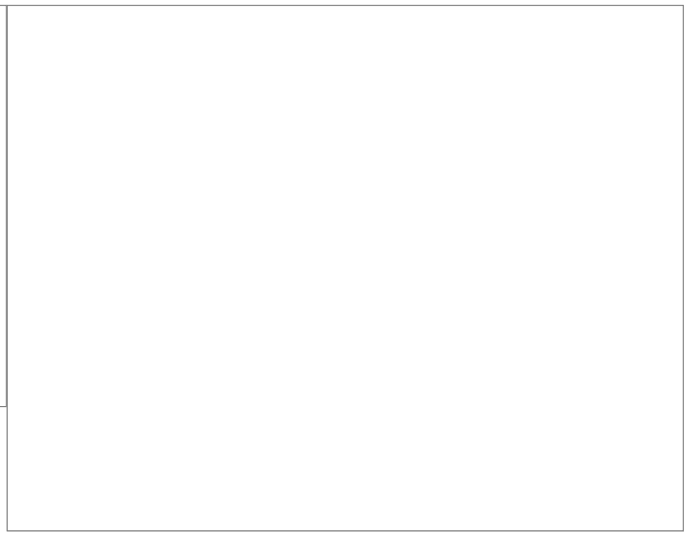
If all goes well, you will receive a confirmation message, as above.

3.8 Redirections

Changing redirections is similar to setting arguments, except that an additional window is used to set the attributes of a redirection.

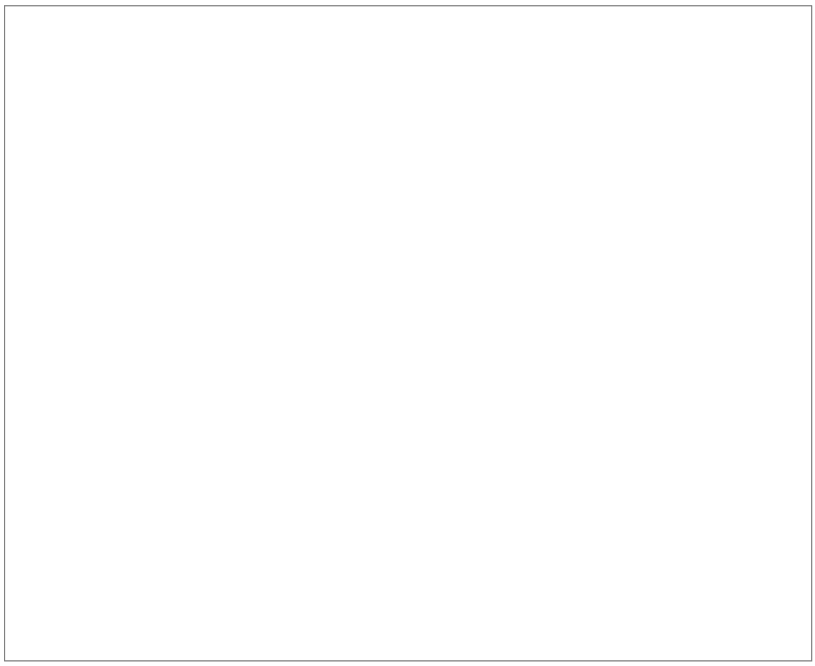
First select the job, and then press the **Redirect I/O** button to yield the screen:

or if there are existing redirections, a screen such as:



You can add a completely new redirection to the new redirection list by clicking on **Add to new** against **New:** or you can add an existing one by clicking on **Add to new** against a listed one.

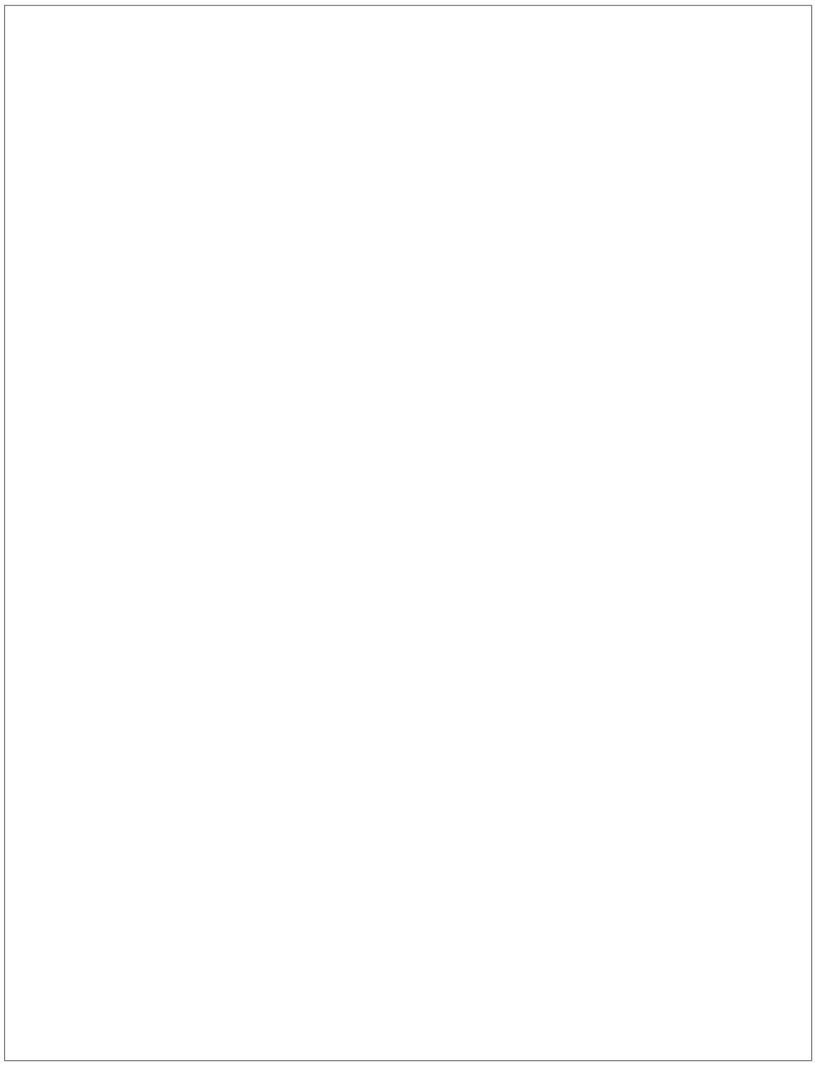
If you add a new one, you will get a new window thus:



The boxes allow you to choose the file descriptor, type of I/O and the file name to be used (or descriptor in one case).

When you have set these, click **Create IO** and a "results" window will appear, thus:

This window is appended to by **Add to New** against an exiting window.



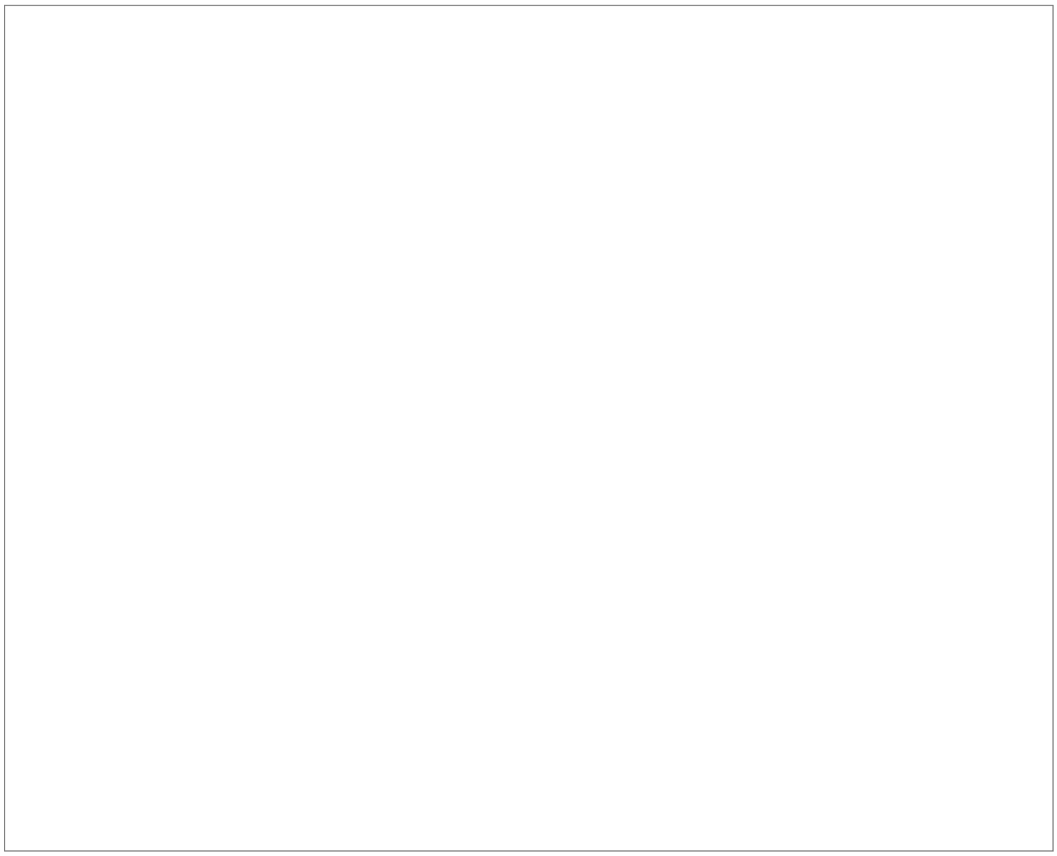
When done, press **Make Changes** on the main screen and the redirections are updated, or press **Cancel** to leave them unchanged.

If all goes well, you will receive a confirmation message, as above.

3.9 Times

You can set the next time to run and repeat options on a job by selecting it and clicking on the **Times** button to give the following window:

The first row lets you choose the first date and time to run (copying the existing time or supplying the current time as a starting time).



You can then set repeat options as per the System Reference Manual. Note however that for "Days Relative to the End" of the month, the notation for the last day of the month is different, you should specify 1 for the last day of the month, 2 for the second last etc.

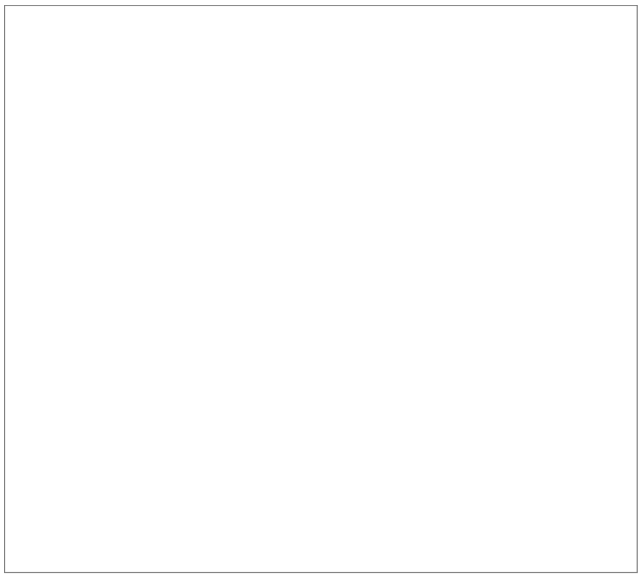
3.10 Conditions

Conditions are set very much in the same way as redirections are. A new list of conditions is constructed by adding from the existing set or adding new conditions.

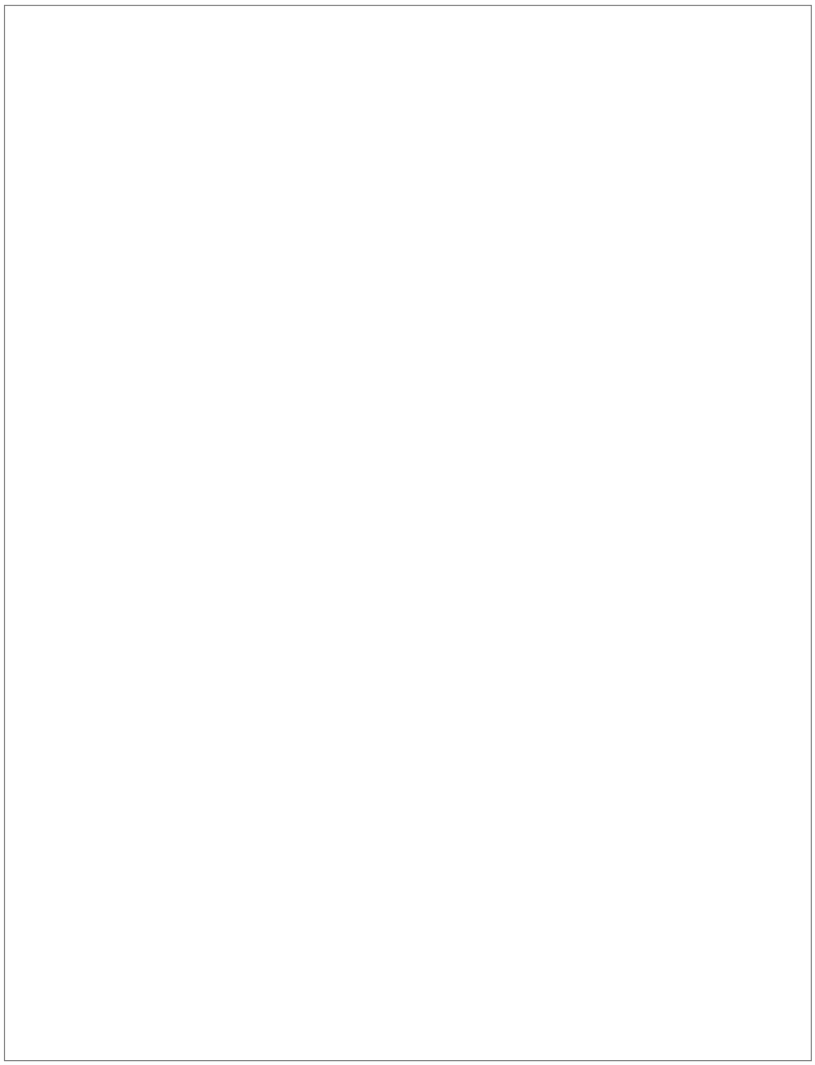
To set conditions on a job, select it and click on **Conditions**.

If the job does not have any conditions as yet, the following window will be displayed:

If there are existing conditions, then the window will look like this:

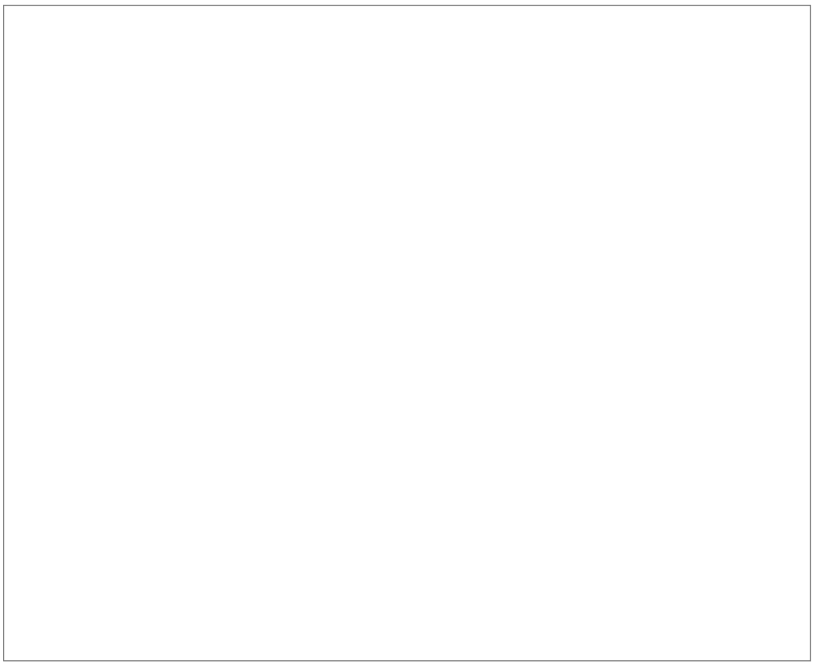


To create the new list of conditions, click on **Add to new** for any existing conditions to be copied to the new list (the order does not matter) and a result window will be created and displayed thus:



Add any new conditions by clicking on the **Add to New** button against **New:** to display this window:

You can click on the selection at the top to get a list of variables which might be appropriate. The middle box gives a list of operations and the final box gives the numeric or string value, followed by the "condition critical" flag.



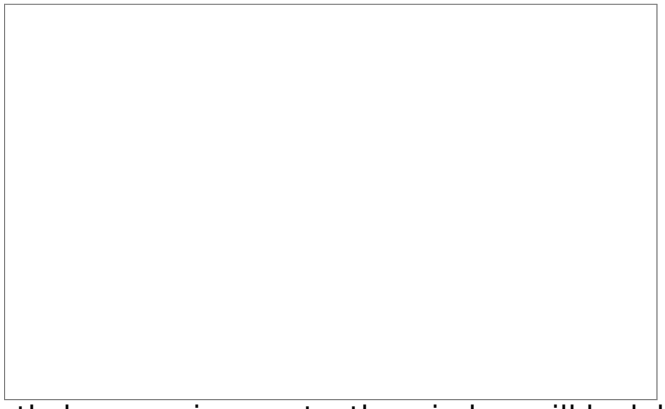
When you have set these, click on **Create Cond** to add it to the results list.

Finally click on **Make Changes** in the main window to copy across the new condition list. If all goes well, you will receive a confirmation message, as above.

3.11 Job Assignments

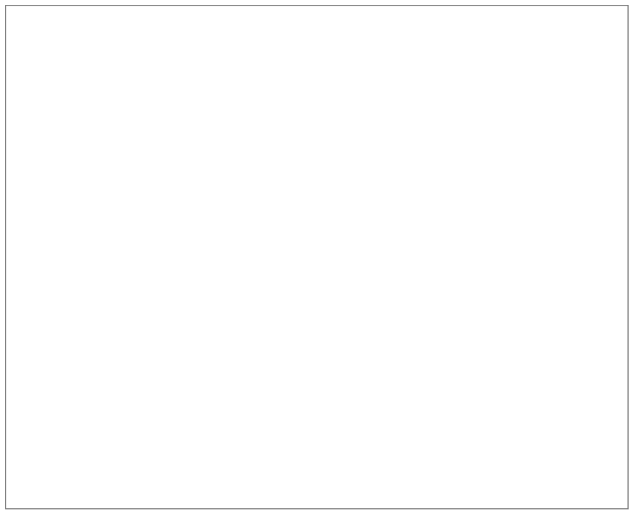
The manner of setting these is very much the same as for conditions.

Select the job and click on **Assignments**. This will display a window such as the following if the job does not have assignments currently:

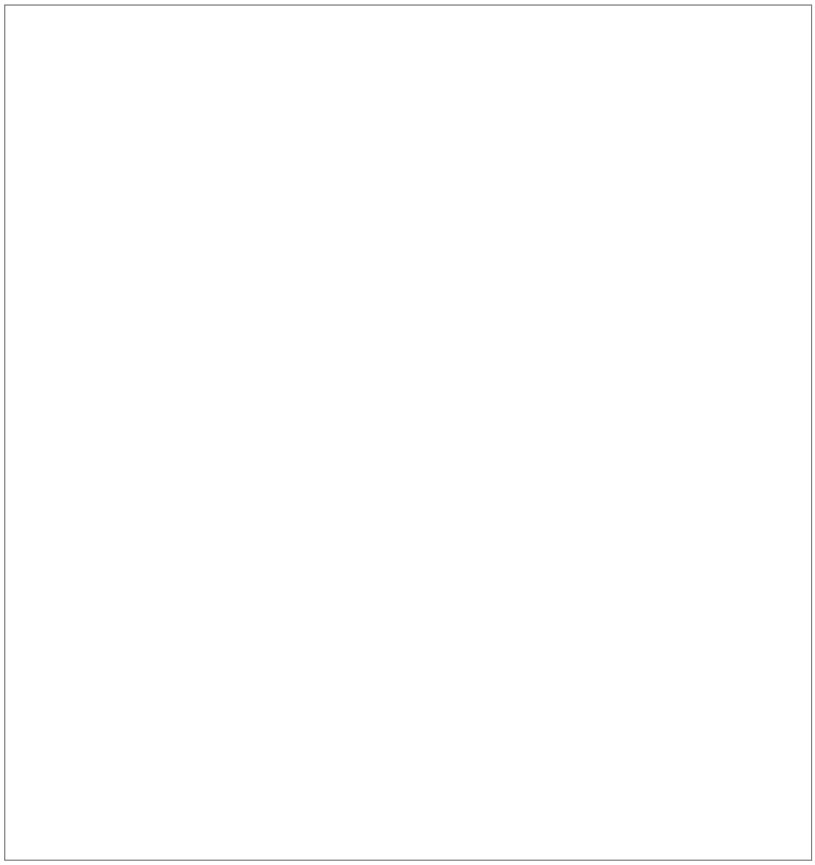


If the job does currently have assignments, the window will look like this:

As with conditions, you can copy the existing assignments the new list by clicking the relevant **Add to new** button and you can add one or more extra ones by clicking the **Add to new** button against **New:**

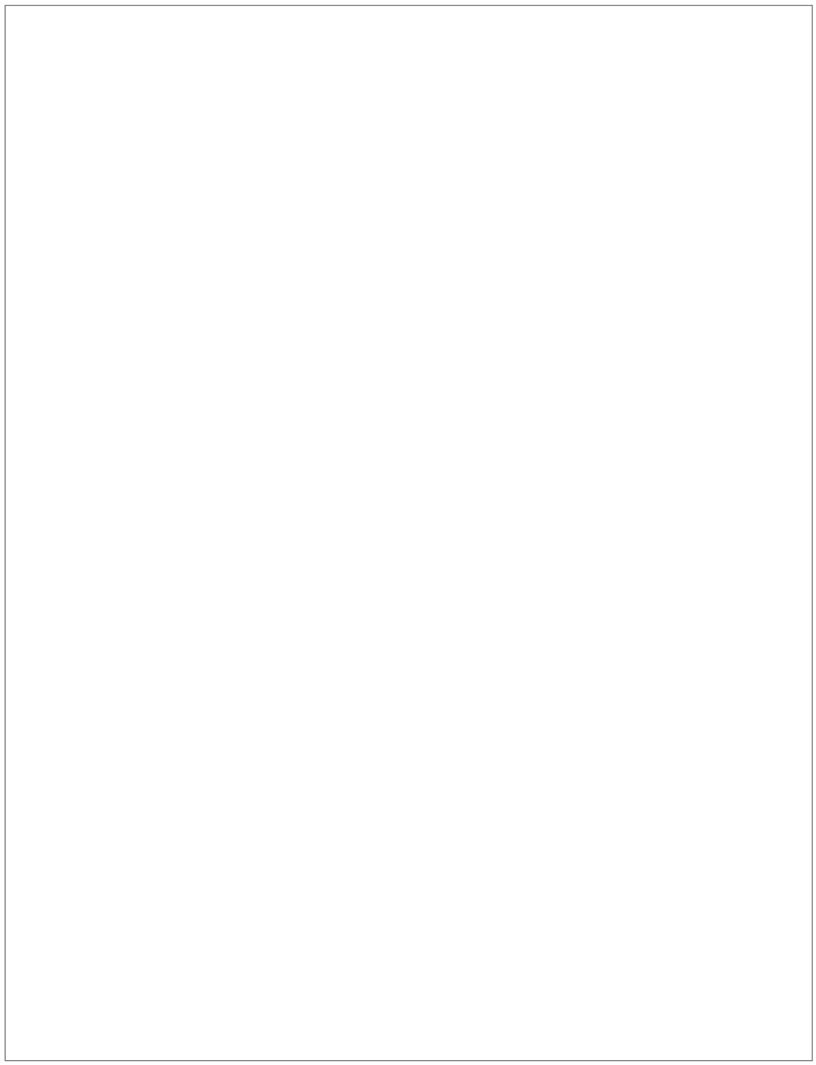


This will generate a window such as



This allows you to scroll through possible variables, and select the relevant condition, action and value. Press **Create Ass** when done. A result window is built up thus:

When you click **Make Changes** on the main window, the result is copied to the job's assignment list. If all goes well, you will receive a confirmation message, as above.



4 Variable Operations

The following operations are available in the web browser interface to administer variables.

In all cases, only one variable at a time should be selected, by clicking on the box at the extreme left of the variable row.

Note that you can clear all the boxes on the job or variable lists by pressing **Clear Marked**.

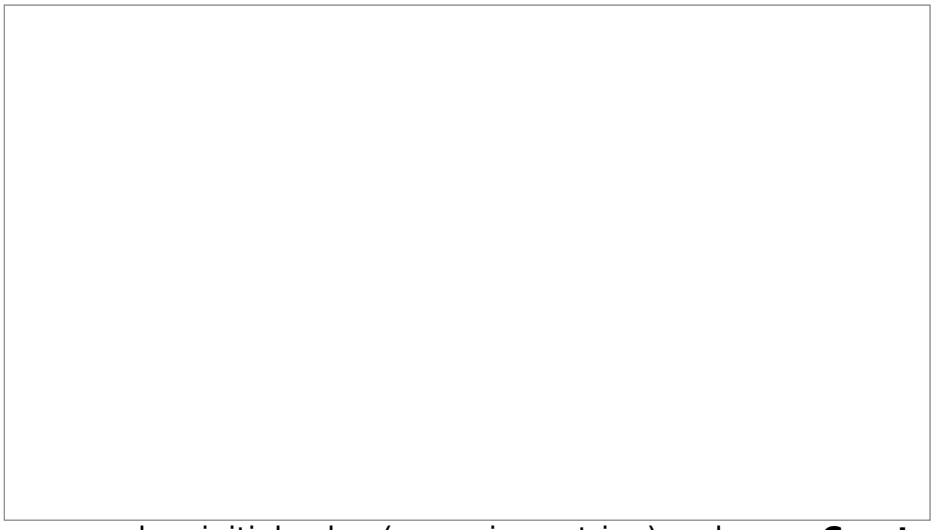
The following operations are documented here.

1. Create variable
2. Delete variable
3. Set variable permissions
4. Set network status
5. Assign value
6. Increment

7. Decrement
8. Set comment

4.1 Create Variable

To create a variable (on the server only), click the **Create** button to get a window as follows:



Fill in the name and an initial value (numeric or string) and press **Create Variable**.

If all goes will you will receive a completion message as follows:

4.2 Delete variable

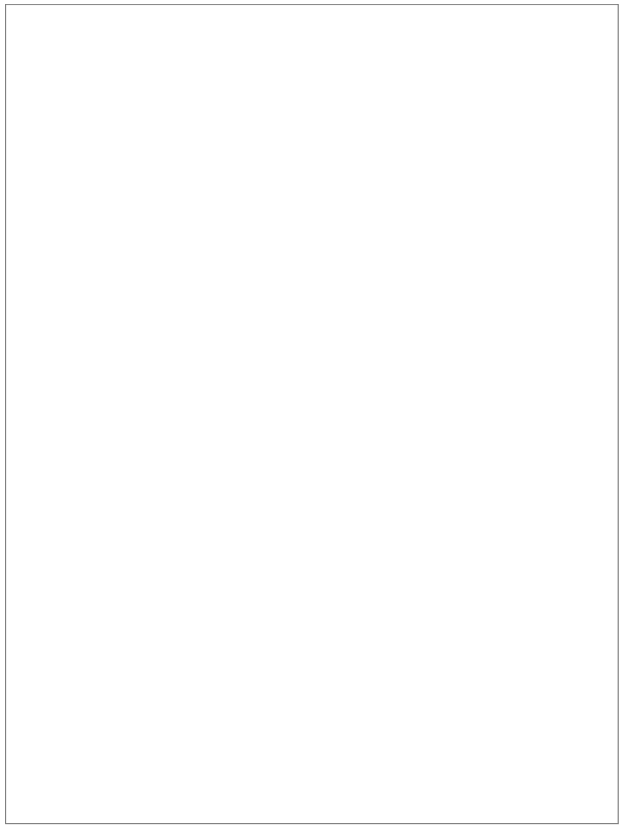
To delete a variable, select it and click on **Delete**.

If all goes well you will get a completion message as above.

4.3 Set variable permissions

Variables are created with a default set of permissions appropriate for the user. To

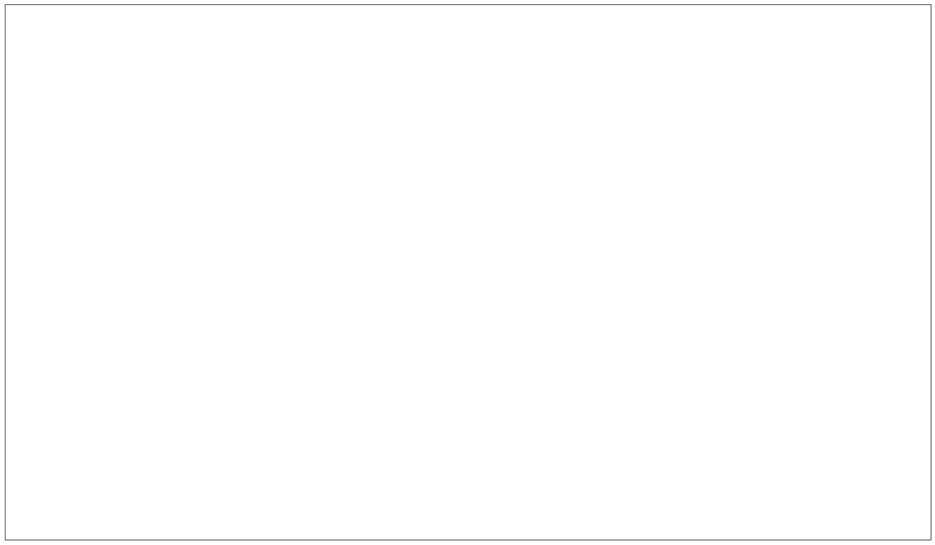
review and change the permissions on a variable, select it and click **permissions**. This will display the following window:



To change permissions, make the required changes and click **Set Permissions**. If all goes well you will get a completion message as above.

4.4 Network Status

To set the network parameters for a variable, select it and click on the **Network** button, to give a window thus:



Make the relevant selection and click **Set export variable**. If all goes well you will get a completion message as above.

4.5 Assign value

To assign a new value to a variable, select it and click on **Assign**.

The following window should appear, enabling you to assign a new numeric or string value.

If all goes well you will get a completion message as above.

4.6 Increment/decrement

These operations increment or decrement a selected variable by one. If all goes well you will get a completion message as above.

4.7 Set Comment

To set the comment on a variable, select it and click on **Comment** to get a window as follow:

Insert a comment string and press **Change Comment**. If all goes well you will get a completion message as above.

5 Installation

5.1 General

Installation of the Web Interface may depend upon the Web Server in use and the locations of the CGI directories. As distributed, the installation puts the various files in the internal program directory for GNUbatch and in two subdirectories for HTML files and CGI binaries.

Most of the HTML files are "template" files used by the CGI binaries and currently only 3 HTML files and the JavaScript routine file need to be placed in the web directories. However the CGI binaries will need to be copied to the CGI binary directory in most cases, as Apache normally will refuse to follow symbolic links.

Some of the files may need adjusting if some or all components are to be placed other than in the top-level directory of the web site.

The local version of the browser assumes that various of the component files of **GNUbatch** are already installed. The remote version of the browser needs a small set of files and possibly the shared version of the API library.

5.2 HTML Files in Web Directory

The following files are located in the web directory. Note that if they are put in a subdirectory, cross-references to them in other files may need to be updated.

These files may be edited and adjusted as required for individual requirements.

5.2.1 gbatch.html

This is the basic frameset for the browser interface, invoking gbjobbutts.html for the button bar and a CGI binary for the list display. This is a symbolic link to gbframe.html for the local browser and gbrframe.html for the remote browser in the HTML files directory. Currently the only difference between the two is that the local browser uses the CGI program btjcgi and the remote uses rbtjcgi.

5.2.2 gbjobbutts.html

This provides the buttons down the left of the screen when displaying jobs. This is a symbolic link to a file of the same name in the HTML files directory.

5.2.3 gbvarbutts.html

This provides the buttons down the left of the screen when displaying variables. This is a symbolic link to a file of the same name in the HTML files directory.

5.2.4 gbjsfunc.js

This provides a library of JavaScript functions used by the interface. This is a symbolic link to a file of the same name in the HTML files directory.

5.3 CGI Binaries

The following CGI binaries are provided by the local browser. The remote version has the names prefixed by "r", thus btjcgi becomes rbtjcgi in the remote browser.

CGI Program	Function
btjcgi	List jobs and parameters
btjccgi	Make changes to jobs
btjdcgi	Delete jobs
btjvcgi	View jobs
btjcrcgi	Create jobs
btvcgi	List variables and parameters
btvccgi	Change variables

Under Apache, they usually have to be installed directly into the cgi-bin directory, as symbolic links on CGI programs are forbidden unless authorised using the FollowSymLinks directive in the Apache configuration file.

5.3.1 Ownership of CGI files

The CGI files are owned by and set-user to user batch for the local interface (in order to access the job and variable shared memory segments) and by and set-user to user root for the remote interface (in order to masquerade as the appropriate user).

The user data is saved with a system group ownership, the CGI files are therefore all group-owned by and set-group id to a system group id such as daemon.

The standard permissions on the CGI files is therefore (octal) `6755` or `rwsr-sr-x` as displayed by ls.

5.4 Parameter File

A parameter file `gbhtml.ini` is placed in the internal programs directory to provide parameters for the browser interface. This is described in more detail in the following chapter.

5.5 HTML Template files

A series of "HTML template files" are used by the CGI programs to generate HTML. Certain parameters, represented by `$Y` and `$Z`, are substituted into the files as the HTML is generated, containing details of job numbers etc.

An "indirect" means of reference is used to ease tuning of the web browser. The CGI programs name a file such as `list_postamble` which is searched for in the parameter file `gbhtml.ini`, giving a file location possibly relative to the location of the parameter file.

5.6 Message File

Messages (error messages in particular) are extracted from the "shell command message file" for **GNUbatch**, `btrest.help`. For the remote browser, this could be cut down to just the messages relevant to the browser, but it hardly seems worth it.

5.7 Host file

Host names may be specified in the `/usr/local/etc/gnubatch.hosts` file for the remote browser, if required, but this need not be included, and if not the gethostname routine is used.

6 Parameters

This chapter describes the parameter file gbhtml.ini in a little more detail.

The file format is based on the types of .ini file which used to be found with Windows etc.

6.1 Location of file

The file is located in the internal program directory which may be relocated using the *master configuration file* /usr/localetc/gnubatch.conf using the keyword SPROGDIR (this applies even to the remote browser, for consistency). Please see the **GNUbatch** reference manual for more information on this.

Files referred to in the parameter file are taken as being relative to the directory in which this file is found if they do not start with a "/", otherwise they are taken literally. For example if the file gbhtml.ini is in /usr/local/var, then a filename of html/listpost.html will be taken to be /usr/local/var/html/listpost.html. This applies after any environment variable expansions have taken place.

6.2 Format of file

The following is a series of extracts from the file. Note that comments start with # and continue to the end of the line.

```
# Files and parameters to control login
# First user key file

userfile=/usr/local/var/gnubatch/html_ufile

# Comment out the following if you do not want a default user
defltuser=guest

# Days after which timeout occurs
usertimeout=7

error_preamble=html/error_pre.html
error_postamble=html/error_post.html

[btjcgi]
headers=y
list_preamble=html/btjcgilist_pre.html
```

The first part of the file, before any [xyz] sections, applies globally to all programs. Sections headed by a program name in [] apply only to that program, overriding the assignments in the global section. In this section, userfile, defltuser, usertimeout, error_preamble and error_postamble are set globally and headers and list_preamble are set only for btjcgi.

Parameters are numbers (as in usertimeout=7), strings (as in defltuser=guest) or file names (as in error_preamble=html/error_pre.html).

Environment variables prefixed with $ (including ones set with "=" or ":" in

/usr/local/etc/gnubatch.conf) and *~username* constructs are expanded in file names only. Environment variables whose values contain further environment variable names are recursively expanded up to 10 times.

Note that the CGI programs never ask for a complete file name, they ask for the parameter named, and this is expanded as required.

6.3 Control parameters

The following parameters have particular importance in configuring the Browser Interface. They should be placed in the global section of the file.

6.3.1 userfile

This locates the file in which user names and host names are encoded to a 32-bit keyword which is used to refer to users.

6.3.2 defltuser

If specified, this gives the name of a default username to access the system unless the user changes identity using "re-login".

6.3.3 deflthost

If specified, this gives the name of a default host to access the system unless the user changes identity using "re-login". This is only relevant for the remote browser, in which case defltuser should also be specified.

6.3.4 usertimeout

This gives the number of days (or alternatively hours and minutes as hh:mm or days, hours and minutes as dd:hh:mm) after which the login record of a user becomes "stale" and the user has to log in again with the correct password.

6.3.5 userrefresh

If set, this indicates that each access to the system should "refresh" the user record, putting off the time after which the user has to log in again.

6.4 Initialising default formats

The initial format of the jobs and variable lists which will be displayed unless the user changes them, or if he/she clicks "reset default" may be defined here. This is done by inserting a parameter of the form format=string in the sections for btjcgi (job listing) or btvcgi (variable listing) or their equivalents rbtjcgi and rbtvcgi for the remote browser.

The string argument in the parameter consists of a sequence of letter pairs. The first letter in each pair is L, R or C, giving the required alignment. The second letter in each pair selects the required field, using the same format codes as are used in the -F options to btjlist and btvlist but without the % signs.

For example the following would specify the default format for job display in btjcgi, as being the standard, but with an initial column of the job start time.

```
[btjcgi]
.......
format=LsLNLULHLILpLLLtLcLP
```