User commands

Viewing information about your cluster

Command	Description
bhosts	Displays hosts and their static and dynamic resources
bmgroup	Displays information about host groups and compute units
blimits	Displays information about resource allocation limits of running jobs
bparams	Displays information about tunable batch system parameters
bqueues	Displays information about batch queues
busers	Displays information about users and user groups
lshosts	Displays hosts and their static resource information
lsid	Displays the current LSF version number, cluster name and master host name
lsinfo	Displays load sharing configuration information
lsload	Displays dynamic load indices for hosts

Monitoring jobs and tasks

Command	Description
bapp	Displays information about jobs attached to application profiles
bhist	Displays historical information about jobs
bjdepinfo	Displays dependency information about jobs.
bjgroup	Displays information about job groups
bjobs	Displays information about jobs
bpeek	Displays stdout and stderr of unfinished jobs
bsla	Displays information about service class configuration for goal-oriented service-level agreement (SLA) scheduling
bstatus	Reads or sets external job status messages and data files

Submitting and controlling jobs

Command	Description
bbot	Moves a pending job relative to the last job in the queue
bchkpnt	Checkpoints a checkpointable job
bgadd	Creates job groups
bgdel	Deletes job groups
bkill	Sends a signal to a job
bmig	Migrates a checkpointable or rerunnable job
bmod	Modifies job submission options
brequeue	Kills and requeues a job
bresize	Releases slots and cancels pending job resize allocation requests.
brestart	Restarts a checkpointed job
bresume	Resumes a suspended job
bstop	Suspends a job
bsub	Submits a job
bswitch	Moves unfinished jobs from one queue to another
btop	Moves a pending job relative to the first job in the queue

bsub command

Syntax

bsub [*options*] *command* [*arguments*]

Options

Option	Description
-ar	Specifies the job is autoresizable
-B	Sends email when the job is dispatched
-H	Holds the job in the PSUSP state at submission
-I | -Ip | -Is	Submits a batch interactive job. -Ip creates a pseudo-terminal. -Is creates a pseudo-terminal in shell mode.
-K	Submits a job and waits for the job to finish
-N	Emails the job report when the job finishes
-r	Makes a job rerunnable
-ul	Passes operating system user shell limits to the execution host (UNIX and Linux only)
-x	Exclusive execution
-app *application_profile_name*	Submits the job to the specified application profile
-b *begin_time*	Dispatches the job on or after the specified date and time in the form [[*month*:]*day*:]:*minute*
-C *core_limit*	Sets a per-process (soft) core file size limit (KB) for all the processes that belong to this job
-c *cpu_time*[/*host_name* | /*host_model*]	Limits the total CPU time the job can use. CPU time is in the form [*hour*:]*minute*
-cwd "*current_working_directory*"	Specifies the current working directory for the job.
-D *data_limit*	Sets per-process (soft) data segment size limit (KB) for each process that belongs to the job
-E "*pre_exec_command* [*arguments* ...]"	Runs the specified pre-exec command on the execution host before running the job
-Ep "*post_exec_command* [*arguments* ...]"	Runs the specified post-exec command on the execution host after the job finishes
-e *error_file*	Appends the standard error output to a file
-eo *error_file*	Overwrites the standard error output of the job to the specified file
-F *file_limit*	Sets per-process (soft) file size limit (KB) for each process that belongs to the job
-f "*local_file op* [*remote_file*]" ...	Copies a file between the local (submission) host and remote (execution) host. *op* is one of >, <, <<, ><, <>
-G *user_group*	Associates job with a specified user group
-g *job_group_name*	Associates job with a specified job group
-i *input_file* | -is *input_file*	Gets the standard input for the job from specified file
-J "*job_name*[*index_list*]%*job_slot_limit*"	Assigns the specified name to the job. Job array *Index_list* has the form *start*[-*end*[:*step*]], and %*job_slot_limit is* the maximum number of jobs that can run at any given time.
-k "*chkpnt_dir* [*chkpnt_period*] [method=*method_name*]"	Makes a job checkpointable and specifies the checkpoint directory, period in minutes, and method
-L *login_shell*	Initializes the execution environment using the specified login shell
-M *mem_limit*	Sets the per-process (soft) memory limit (KB)
-m "*host_name* [@*cluster_name*][[!] | +[*pref_level*]] | *host_group*[[!] |+[*pref_level*]] | *compute_unit*[[!] |+[*pref_level*]]..."	Runs job on one of the specified hosts. Plus (+) after the names of a host or group indicates a preference. Optionally, a positive integer indicates a preference level with higher numbers indicating a greater preference.
-n *min_proc*[,*max_proc*]	Specifies the minimum and maximum numbers of processors required for a parallel job
-o *output_file*	Appends the standard output to a file
-oo *output_file*	Overwrites the standard output of the job to the specified file
-P *project_name*	Assigns job to specified project
-p *process_limit*	Limit the number of processes for the whole job
-Q "[*exit_code* …] [EXCLUDE(*exit_code* …)]"	Specifies automatic job requeue exit values and exclusive job requeue exit values
-q "*queue_name* ..."	Submits job to one of the specified queues
-R "*res_req*" [-R "*res_req*" …]	Specifies host resource requirements
-rnc *resize_notification_cmd*	Specifies full path of an executable invoked on the first execution host when the job allocation is modified
-S *stack_limit*	Sets a per-process (soft) stack segment size limit (KB) for each process that belongs to the job
-s *signal*	Send *signal* when a queue-level run window closes
-sla *service_class_name*	Specifies the service class where the job is to run
-sp *priority*	Specifies user-assigned job priority allowing users to order their jobs in a queue
-T *thread_limit*	Sets the limit of the number of concurrent threads for the whole job
-t *term_time*	Specifies the job termination deadline in the form [[*month*:]*day*:]*hour*:*minute*
-U *reservation_ID*	Use advance reservation created with brsvadd
-u *mail_user*	Sends mail to the specified email address
-v *swap_limit*	Set the total process virtual memory limit (KB) for the whole job
-W *run_time*[/*host_name* | /*host_model*]	Sets the run time limit of the job in the form [*hour*:]*minute*
-w '*dependency_expression*'	Places a job when the dependency expression evaluates to TRUE
-wa '*signal*'	Specifies the job action to be taken before a job control action occurs
-wt '[*hour*:]*minute*'	Specifies the amount of time before a job control action occurs that a job warning action is to be taken
-Zs	Spools a command file for the job to the directory specified by the JOB_SPOOL_DIR in lsb.params
-h	Prints command usage to stderr and exits
-V	Prints LSF release version to stderr and exits

Platform

Release date: September 2008

www.platform.com
doc@platform.com
Last modified: February 26 2009