m21info (mpinfo) Functionality

m21info (mpinfo) Functionality

Table of Contents

Chapter 1. Overview

This document describes the program that is used to examine diagnose and manage a running M21 system.

mpinfo is a comprehensive system management utility for the M21 system. This document describes the functionality of the utility. **mpinfo** is called using the shell script **m21info** .

Table 1-1.

help	Show help information for a command
lock	Lock system locks
quit	Quit the program
release	Release resources
repeat	Repeat a command
resume	Cause a job that was suspended to resume execution
set	Set internal parameters and system information
show	Show information
shutdown	Shut down the M21 database
step	Single step a job one command at a time
suspend	Suspend a job
wakeup	Wake up a job waiting on a lock
unlock	Unlock system locks
zap	Zap a job

Command Examples

Command words can be abbreviated to the least ambiguous number of characters. Multiple commands can be entered at one time, these must be separated by a semicolon surrounded by white space.

```
show jobs ; show locks
set repeat on ; zap job 9
```

Help Examples

Type help and the command for more detail information.

```
help set
help show
```

Chapter 2. help

More detailed information about a command can be obtained by typing help followed by the command name.

Examples

```
help show
help show jobs
```

Chapter 3. lock

Lock a system resource.

lock bp

lock bp [number] NNNN [exclusive] [shared] [wait]

Lock a buffer pointer. The lock will be locked shared without waiting by default.

Table 3-1.

number NNNN	Buffer pointer number to lock.
exclusive	Lock the buffer pointer exclusively.
shared	Lock the buffer pointer in shared mode , this is the default.
wait	Wait to get the lock, the default is not to wait.

Examples

```
lock bp 10
lock bp 11 exclusive
lock bp number 12 wait
lock bp 13 exclusive wait
```

lock general

lock general GENERAL-SYSTEM-LOCK [exclusive] [shared] [wait]

Lock a **GENERAL-SYSTEM-LOCK** . By default the lock will be locked shared without waiting.

Table 3-2. GENERAL-SYSTEM-LOCK Values:

bitmap	bpoold	bpoolg
bpoolh	copybuf	global
hogsys	jnlupdate	jnlwaiter
lockmgr	mbilsmap	msgbuf
quiesce	setdev	shrmem
svector	symbuf1	wvector

Table 3-3.

exclusive	*Lock the* **GENERAL-SYSTEM-LOCK** *exclusively.*
shared	Lock the **GENERAL-SYSTEM-LOCK** shared, this is the default.
wait	Wait to get the lock, the default is not to wait.

Examples

```
lock general quiesce
lock general quiesce exclusive
lock general svector wait
```

lock hashqueues

lock hashqueue [number] NNNN [exclusive] [shared] [wait]

Lock a hash queue. By default the lock will be locked shared without waiting.

Table 3-4. lock hashqueue options

number NNNN	Hash queue number to lock.
exclusive	Lock the hash queue exclusively.
shared	Lock the hash queue shared, this is the default.
wait	Wait to get the lock, the default is not to wait.

Examples

```
lock hashqueue 10
lock hashqueue 11 exclusive
lock hashqueue number 12 wait
lock hashqueue 13 exclusive wait
```

lock volume

lock volume [number] NN [exclusive] [shared] [wait]

Lock a volume. By default the lock will be locked shared without waiting.

Table 3-5. lock volume options

number NN	Volume number to lock.
exclusive	Lock the volume exclusively.
shared	Lock the volume shared, this is the default.
wait	Wait to get the lock, the default is not to wait.

Examples

```
lock volume 0
lock volume 1 exclusive
lock volume number 2 wait
lock volume 3 exclusive wait
```

Chapter 4. quit

Get out of **mpinfo** . Releases any locks that have been locked using the lock command. DOES NOT resume jobs suspended by the suspend job command.

Chapter 5. release

release ddb

release lock

Releases device descriptors (ddbs) or locks owned by a M(UMPS) job..

release ddb

release ddb [number] NNN

Release a ddb owned by a job. After showing the current state of the ddb you will be prompted if you really want to release the ddb.

Examples

```
release ddb 10
release ddb number 11
```

release locks

release locks [job] NNN [mumps] [system]

Release both MUMPS and SYSTEM locks by default for a job. If either the keyword **mumps** or **system** is specified without the other, then only the lock type specified will be released.

Examples

```
release locks 10
release locks job 11 mumps
release locks 12 system
```

Chapter 6. repeat

repeat COMMAND

Repeat a COMMAND, where command is a valid command e.g. **show jobs** . The repeat count and delay between executing iterations of the command is controlled by the **repeatcount** and **repeatdelay** values. This command is often used when specifying an input file of commands to execute.

SEE ALSO:

set repeatcount

set repeatdelay

Examples

```
repeat show jobs flags
repeat show bp inuse
```

Chapter 7. resume

resume job

Resume a job that has been suspended from **mpinfo** using the **suspend** command.

resume job

resume job [number] NNN

This will cause a job that has been suspended by the **suspend job** command to resume executing.

Examples

```
resume job 10
resume job number 11
```

Chapter 8. set

Used to set options within **mpinfo** and to send messages to the running processes of the M21 to system to request that they carry out activities such as logging statistics.

set bilrequest

set bilrequest BIL-REQUEST-BIT-NAME [off]

Set the **BIL-REQUEST-BIT-NAME** to cause the Before Image Journal program **mpbil** to perform the given request.

BIL-REQUEST-BIT-NAME:

Table 8-1.

countfree	*Request* **mpbil** *to request the disk writers to read the map blocks and count the free blocks.*
quiesce	*Request* **mpbil** *to do a quiesce.*
showstats	*Request* **mpbil** *to log the general statistics in the log file.*
sustain	*Request* **mpbil** *to do an extended quiesce.*
trace0	*Request* **mpbil** *to disable tracing.*
trace1	*Request* **mpbil** *to Enable tracing at level 1.*
trace2	*Request* **mpbil** *to Enable tracing at level 2.*
trace3	*Request* **mpbil** *to Enable tracing at level 3.*
wakeup	*Request* **mpctl** *to wakeup* **mpbil** .
off	*This will cause the* **BIL-REQUEST-BIT-NAME** *to be cleared, the default is to set the bit.*

If **wakeup** is not specified the request may not be looked at for up to one minute. This is because **mpctl** will wake up **mpbil** by default once a minute for it to check things, but **mpctl** checks every second to see if it needs to wake up **mpbil** .

Examples

```
set bilrequest quiesce
set bilrequest quiesce sustain
set bilrequest quiesce wakeup
set bilrequest showstats wakeup
set bilrequest sustain off
set bilrequest trace1 wakeup
```

set ctlrequest

set ctlrequest CTL-REQUEST-BIT-NAME [off]

Set the **`CTL-REQUEST-BIT-NAME`** to cause the control program **mpctl** to perform the given request.

CTL-REQUEST-BIT-NAME:

Table 8-2.

showbpa	Request **mpctl** to log the buffer pool allocation infotmation in the log file.
showdbvi	*Request* **mpctl** *to log the database volume information to the log file.*
showparms	*Request* **mpctl** *to log the current configuration parameters to the log file.*
showstats	*Request* **mpctl** *to log the system status message in the log file.*
systemstats	*Request* **mpctl** *to request* **mpbil** *, all the* **mpdsk** *and* **mpjnl** *to log their statistics in the log file.*
trace0	*Request* **mpctl** *to disable tracing.*
trace1	*Request* **mpctl** *to Enable tracing at level 1.*
trace2	*Request* **mpctl** *to Enable tracing at level 2.*
trace3	*Request* **mpctl** *to Enable tracing at level 3.*
off	*This will cause the* **CTL-REQUEST-BIT-NAME** *to be cleared, the default is to set the bit.*

Examples

```
set ctlrequest showbpa
set ctlrequest showdbvi showstats
set ctlrequest showstats
set ctlrequest trace0
set ctlrequest trace1
```

set debug

set debug all,1,2,3 on,off

Turn on or off 3 debugging bits on the shared memory. These bits are tested by other programs to print or log debugging information.

Examples

```
set debug 1 on
set debug 3 off
set debug all on
```

set dskrequest

set dskrequest DSK-REQUEST-BIT-NAME [all] [[number] NN] [off] [volume XX]

Set the **DSK-REQUEST-BIT-NAME** to cause one of the disk writer programs **mpdsk** to perform the given request.

DSK-REQUEST-BIT-NAME:

Table 8-3.

countfree	*Request* **mpdsk** *to read the map blocks and count the free blocks.*
showstats	*Request* **mpdsk** *to log general statistics in the log file.*
trace0	*Request* **mpdsk** *to disable tracing.*
trace1	*Request* **mpdsk** *to Enable tracing at level 1.*
trace2	*Request* **mpdsk** *to Enable tracing at level 2.*
trace3	*Request* **mpdsk** *to Enable tracing at level 3.*
updblk0	Request the **mpdsk** to update block 0 for the volume it writes if it is a master writer.
wakeup	*Request* **mpctl** *to wakeup* **mpdsk** *.*
all	*Set or clear the* **DSK-REQUEST-BIT-NAME** *for all disk writers.*
number NN	*Set or clear the* **DSK-REQUEST-BIT-NAME** *for disk writer* **NN** *only.*
off	*This will cause the* **DSK-REQUEST-BIT-NAME** *to be cleared, the default is to set the bit.*
volume XX	*Set or clear the* **DSK-REQUEST-BIT-NAME** *for disk writers for volume* **XX** *only.*

If **wakeup** is not specified the request may not be looked at for a few seconds. This

is because **mpctl** will wake up the disk writers every few seconds to look for buffers that have aged. If there are not any modified buffers the disk writers will not get woken up, but **mpctl** checks once a second to see if it needs to wake up any of the disk writers.

Examples

```
set dskrequest trace1 wakeup
set dskrequest updblk0 1
set dskrequest updblk0 wakeup 2
set dskrequest updblk0 wakeup number 3
set dskrequest updblk0 number 4 off
set dskrequest showstats all
set dskrequest showstats volume 2
```

set input

set input file FILE-NAME

set input none

Set the input to be read from **FILE-NAME** . When the end of file is reached the input will automatically switch back to the terminal. The second form is used to cause the input to go to the terminal again, closing **FILE-NAME** .

Examples

```
set input file /tmp/showlots
set input none
```

set jnlrequest

set jnlrequest JNL-REQUEST-BIT-NAME [off]

Set the **`JNL-REQUEST-BIT-NAME`** to cause the after image journal program **mpjnl** to perform the given request.

JNL-REQUEST-BIT-NAME:

Table 8-4.

flush	*Request* **mpjnl** *to flush the current buffer without waiting for it to age or fill up and move to the next buffer and block in the journal files.*
pause	*Request* **mpjnl** *to pause until told to continue. This should be used with care.*

showstats	*Request* **mpjnl** *to log general statistics in the log file.*
syncf1f2	*Request* **mpjnl** *to synchronize the journal file by copying file 1 to file 2.*
syncf2f1	*Request* **mpjnl** *to synchronize the journal file by copying file 2 to file 1.*
updblk0	*Request* **mpjnl** *to update block 0 of the journal files.*
trace0	*Request* **mpjnl** *to disable tracing.*
trace1	*Request* **mpjnl** *to Enable tracing at level 1.*
trace2	*Request* **mpjnl** *to Enable tracing at level 2.*
trace3	*Request* **mpjnl** *to Enable tracing at level 3.*
wakeup	*Request* **mpctl** *to wakeup* **mpjnl** .
off	*This will cause the* **JNL-REQUEST-BIT-NAME** *to be cleared, the default is to set the bit.*

If **wakeup** is not specified the request may not be looked at for a few seconds. This is because **mpctl** will wake up **mpjnl** every few seconds to check the current journal buffer age, but **mpctl** checks once a second to see if it needs to wake up **mpjnl** .

Examples

```
set jnlrequest flush
set jnlrequest showstats wakeup
set jnlrequest syncf1f2 wakeup
set jnlrequest syncf2f1 off
```

set output

set output file FILE-NAME [append]

set output none

Set the output of the show command to go to file **FILE-NAME** . If the keyword append is specified the output will be appended to the file if it exists, otherwise the file will be truncated when opened. The second form is used to cause the output to go to the terminal again, closing **FILE-NAME** .

Examples

```
set output file /tmp/problem1
set output file /tmp/problem2 append
set output none
```

set page

set page NN

Set the page length for output that may require more than one screen to show all of the information requested. When the number of lines specified is reached the output will stop and **CONTINUE** will be displayed at the bottom of the display. The default is 24 if the environment variable **LINES** is not set. If page size is set to 0 then output will not stop until completed.

Examples

```
set page 24
set page 66
set page 0
```

set repeat

set repeat on | off

By default entering a **return** at the command prompt will produce a message about missing input. If **repeat** is set to **on** , then the last command will be repeated **return** is entered.

Examples

```
set repeat on
set repeat off
```

set repeatcount

set repeatcount N

Set the number of times that a command should be executed when preceded by the **repeat** command. This is often used when an input file is being accessed to execute commands.

Examples

```
set repeatcount 5
set repeatcount 10
```

set repeatdelay

set repeatdelay N

Set the number of seconds to delay between executing commands preceded by the **repeat** command. This is often used when an input file is being accessed to execute commands.

Examples

```
set repeatdelay 2
set repeatdelay 5
```

set swdrequest

set swdrequest SWD-REQUEST-BIT-NAME [off]

Set the **SWD-REQUEST-BIT-NAME** to cause the system watchdog program **mpswd** to perform the given request.

SWD-REQUEST-BIT-NAME:

Table 8-5.

showstats	*Request* **mpswd** *to log general statistics in the log file.*
trace0	*Request* **mpswd** *to disable tracing.*
trace1	*Request* **mpswd** *to Enable tracing at level 1.*
trace2	*Request* **mpswd** *to Enable tracing at level 2.*
trace3	*Request* **mpswd** *to Enable tracing at level 3.*
wakeup	*Request* **mpctl** *to wakeup* **mpswd** .
off	*This will cause the* **SWD-REQUEST-BIT-NAME** *be cleared, the default is to setthe bit.*

If **wakeup** is not specified the request may not be looked at for a few seconds. This is because **mpswd** will wake up every few seconds to check if the ports file has been modified, but **mpctl** checks once a second to see if it needs to wake up **mpswd** .

Examples

```
set swdrequest showstats
set swdrequest showstats wakeup
```

set tracing

set tracing job NNN [[level] N]

Set trace level for a job. If the level is not specified it will default to 0. When tracing is enabled for a job, the tracing information is written to a trace file. The file is located in the **/tmp** directory as **T.mpmse.PID** where **PID** is the jobs process ID.

WARNING: Enabling tracing for a job can severely impact the jobs performance.

Examples

```
set tracing job 10
set tracing job 11 level 2
set tracing job 12 3
```

Chapter 9. show

Show detailed system information.

show bilrequest

show bilrequest

Show the current request for the Before Image Journal program **mpbil** .

Table 9-1.

COUNTFREE	Quiesce and then have the disk writers count the free blocks.
QUIESCE	Quiesce the database.
SHOWSTATS	*Show the general statistics for* **mpbil** .
SUSTAIN	Extend the quiesce by not enabling database writes until requested.
TRACE0	Disable tracing.
TRACE1	Enable tracing at level 1.
TRACE2	Enable tracing at level 2.
TRACE3	Enable tracing at level 3.
WAKEUP	*The* **mpctl** *process needs to wake up* **mpbil** .

show_bilstatus

show bilstatus

Show the current status of the Before Image Journal program **mpbil** .

Table 9-2.

ACTIVE	*The* **mpbil** *process is running.*
COUNTFREE	*The* **mpbil** *process is waiting for the disk writers to count the free blocks.*
DBRECOV	*The* **mpbil** *process is performing a database recovery roll back.*
EXTENDED	*The* **mpbil** *process is in an extended quiesce. The quiesce has completed, but writing is still disabled for jobs.*
QFAILED	*The* **mpbil** *process failed to quiesce on the last try.*
QUEISCE	*The* **mpbil** *process is doing a quiesce.*

READY	*The* **mpbil** *process is ready, initialization is complete.*
TRACE1	*The* **mpbil** *process is logging trace level 1 information to a trace log file.*
TRACE2	*The* **mpbil** *process is logging trace level 2 information to a trace log file.*
TRACE3	*The* **mpbil** *process is logging trace level 3 information to a trace log file.*
WAITING	*The* **mpbil** *process is waiting on its wait queue.*
WRITING	*The* **mpbil** *process is writing a block to the bil file.*

Examples

```
show bilstatus
```

show block

show block NNNNNNN [character] [decimal] [dump] [from NNNN] [hex] [list] [octal] [to NNNN]

If block **NNNNNNN** can be found in the buffer pool then the buffer pointer will be displayed, otherwise block **NNNNNNN** will be read from disk. In either case the block type and other information will be displayed. The output will vary depending on the block type.

Table 9-3.

character	When the `dump` keyword is used each byte will be displayed as a character.
decimal	*When the* **dump** *keyword is used each byte will be displayed as a decimal value from 0 to 255.*
dump	*Causes the contents of the block to be displayed byte by byte. See* **character** *,* **decimal** *,* **hex** *, and* **octal** *for more detail. All formats can be requested at the same time, the default is* **character** *.*
from NNNN	*When the* **dump** *keyword is used* **NNNN** *will be the starting offset within the block to dump, default offset is* **0** *.*
hex	*When the* **dump** *keyword is used each byte will be displayed as a hex value from 0 to FF.*

list	Causes the contents of the block to be listed based on the type of block.
octal	*When the* **dump** *keyword is used each byte will be displayed as an octal value from 0 to 377.*
to NNNN	*When the dump keyword is used* **NNNN** *will be the ending offset within the block to dump, the default is the size of the block.*

Examples

```
show block 10
show block 10 list
show block 20 dump
show block 30 dump hex
show block 40 dump from 100 to 200 octal decimal
```

show bps

show bps [NNNN] [access] [all] [data] [date] [general] [hog] [inuse] [jobs] [modified] [number NNNN] [pattern PAT] [summary] [used] [volume NN]

Show buffer pointer, bp pool, block number, block type, volume, use count, modified, etc. The default is to show all the buffer pointers as if the keyword `all` was specified.

Table 9-4.

access	Show last accessed time and age.
all	Show all buffer pools.
data	Show bps in the data buffer pool.
date	Show the bp modification time and the date when the block was written to the database volume.
general	Show bps in the general buffer pool.
hog	Show bps in the hog buffer pool.
jobs	Show only bps that have a lock count not equal 0, and list jobs that have the bp lock on their Process Lock Table.
inuse	Show only bps that have a lock count not equal 0.
modified	Show only bps that are marked modified.
number NNNN	*Show information for bp* **NNNN** *only.*

pattern PAT	Show only bps that output would match the pattern.
summary	Give summary information only.
used	Show only bps that have been used at some time.
volume NN	*Show only bps that contain a block in volume* **NN** *.*

Examples

```
show bps data
show bps date
show bps general hog
show bps 10
show bps inuse
show bps hog inuse
show bps data modified
show bps summary
show bps data used
show bps pattern %UDE
show bps volume 1
```

show clock

show clock

Shows the system time and M21 time. Also shows the difference if there is any.

show ddbs

show ddbs [all] [[number] NNN] [class all,dynamic,physical,software,virtual] [group all,hostspool,port,tape,view] [pattern PAT] [type all,instrument,modem,printer,special,terminal]

Show device descriptor type, owning job, and other information.

Table 9-5.

all	Show even unassigned ddbs. The default is to only show ddbs that are assigned or in use in the case of the dynamic port class.
class all	Show all classes.
class dynamic	Show ports that are used as dynamic ports.

class physical	Show ports that are used for physical connections.
group all	Show all groups.
group hostspool	Show host spool ddbs.
group port	Show port ddbs.
group tape	Show tape ddbs.
group view	Show view ddbs.
number NNNN	*Show information for ddb* **NNNN** *only.*
pattern PAT	Show only ddbs where output would match the pattern.
type instruments	Show ports that are used as instruments.
type modem	Show ports that are used as modems.
type printer	Show ports that are used as printers.
type terminals	Show ports that are used as terminals.

Examples

```
show ddbs
show ddbs all
show ddbs 10
show ddbs class physical
show ddbs class physical pattern tty101
show ddbs class virtual pattern 139.177.254.201
show ddbs group tape
show ddbs group port type printer
```

show dskstatus

show dskstatus [[number] NN] [volume VV]

Show the current status of the disk writers. If **NN** is specified then only information for disk writer **NN** will be shown. If the keyword volume is specified followed by a volume number then information for all disk writers for volume **VV** will be shown. The default is to show the status for all disk writers.

Table 9-6.

ACTIVE	*The* **mpdsk** *process is running.*
COUNTFREE	*The* **mpdsk** *process is counting the free blocks.*
MASTER	*The* **mpdsk** *process is a master writer.*

READING	*The* **mpdsk** *process is reading a block form the volume.*
READY	*The* **mpdsk** *process is ready, initialization is complete.*
TRACE1	*The* **mpdsk** *process is logging trace level 1 information to a trace log file.*
TRACE2	*The* **mpdsk** *process is logging trace level 2 information to a trace log file.*
TRACE3	*The* **mpdsk** *process is logging trace level 3 information to a trace log file.*
UPDATING	*The* **mpdsk** *process is updating block 0 for the volume.*
WAITING	*The* **mpdsk** *process is waiting on its wait queue.*
WRITING	*The* **mpdsk** *process is writing a block to the volume.*

Examples

```
show dskstatus
show dskstatus 2
show dskstatus volume 2
```

show hashqueues

show hashqueues [all] [number NNNN] [used]

The default is to show only the summary information.

Table 9-7.

all	Shows the number of bps on each hash queue.
number NNNN	*Show the bps on hash queue* **NNNN** *. Displays the bp number, flags, block number and block type.*
used	Show only hash queues with a count greater than 0.

Examples

```
show hashqueues
show hashqueues all
show hashqueues number 10
```

```
show hashqueues used
```

show information

show information [all] [bil] [general] [journal] [volumes [NN]]

The default is as if the keyword **general** had been specified which will show general information about M21, maximum number jobs and ports, block size, shared memory size, etc .

Table 9-8.

all	*Show information as if keywords (* **bil** *,* **general** *,* **journal** *, and* **volumes** *) have been specified.*
bil	Show information about the Before Image Logging file location, size, etc.
journal	Show information about the Journal file location, size, etc.
stats bps	Show buffer pool statistics.
stats rtc	Show routine cache statistics.
volumes [NN]	*Show information about database Volume file location, size, etc. If* **NN** *is specified then only the information for volume* **NN** *will be displayed.*

Examples

```
show information
show information bil
show information journal volumes
show information stats bps
show information volumes 5
show information all
```

show jnlrequest

show jnlrequest

Show the current request for the after image journal program **mpjnl** as follows:

Table 9-9.

FLUSH	Flush the current block to disk even if it is not full or aged.

PAUSE	*Pause writing blocks while* **dbbkrs** *updates some information after completing a journal backup.*
SHOWSTATS	*Show the general statistics for* **mpjnl** .
SYNCF1F2	Synchronize journal files by copying blocks from file 1 to file 2.
SYNCF2F1	Synchronize journal files by copying blocks from file 2 to file 1.
TRACE0	Disable tracing.
TRACE1	Enable tracing at level 1.
TRACE2	Enable tracing at level 2.
TRACE3	Enable tracing at level 3.
UPDBLK0	Update block zero with the current quiesce number and update the usage information.
WAKEUP	*The* **mpctl** *process needs to wake up* **mpjnl** .

show jnlstatus

show jnlstatus

Shows the current status of the after image journal program **mpjnl** as follows:

Table 9-10.

ACTIVE	*The* **mpjnl** *process is active.*
F1ACCESS	*The* **mpjnl** *process can access to journal file 1.*
F1READ	*The* **mpjnl** *process is reading a block from journal file 1.*
F1WRITE	*The* **mpjnl** *process is writing a block to journal file 1.*
F1UPDBLK0	*The* **mpjnl** *process will update block zero when needed for journal file 1.*
F2ACCESS	*The* **mpjnl** *process can access to journal file 2.*
F2READ	*The* **mpjnl** *process is reading a block from journal file 2.*
F2WRITE	*The* **mpjnl** *process is writing a block to journal file 2.*

F2UPDBLK0	*The* **mpjnl** *process will update block zero when needed for journal file 2.*
FMODE1	*The* **mpjnl** *process is using on 1 journal file.*
FMODE2	*The* **mpjnl** *process is using on 2 journal files.*
NEEDSYNC	The journal files need synchronizing.
OVERWRITE	*The* **mpjnl** *process is over writing oldest data in the journal files.*
READY	*The* **mpjnl** *process is ready, initialization has completed.*
STOPPED	*The* **mpjnl** *process has stop journaling.*
SYNCING	*The* **mpjnl** *process is synchronizing the journal files.*
TRACE1	*The* **mpjnl** *process is logging trace level 1 information to a trace log file.*
TRACE2	*The* **mpjnl** *process is logging trace level 2 information to a trace log file.*
TRACE3	*The* **mpjnl** *process is logging trace level 3 information to a trace log file.*
WAITING	*The* **mpjnl** *process is waiting on its wait queue.*

show jobs

show jobs [activity][dead] [devices] [flags] [memory] [[number] NNN] [pattern PAT] [pid P] [plt] [sort commands,cputime,id,elcount,ewcount,ewtime,lreads, preads,routine,runtime,slcount,swcount,swtime] [status] [waiting]

Show the job number, process id, MUMPS UCI, MUMPS routine, etc.

Table 9-11.

activity	Show the activity of jobs, i.e. number of commands executed, how long the job has been running, and the amount of CPU time the job has used.
dead	Show only dead jobs.
device	*Show principle device, current device and any other device the job has opened. Mutually exclusive with keywords* **flags** *,* **memory** *, and* **status** *.*

flags	*Show flag information [ZJOB, ZWAIT, etc...]. Mutually exclusive with keywords* **device** *,* **memory** *, and* **status** *.*
memory	*Show symbol table memory information. Mutually exclusive with keywords* **device** *,* **flags** *, and* **status** *.*
number NNN	*Show information for job number* **NNN** *only.*
pattern PAT	*Show only jobs that have information that match the pattern* **PAT** *. This can be used to match device number, status, etc.*
pid P	*Show only job with pid* **P** *.*
plt	*Show Process Lock Table information. Must specify job number* **NNN** *.*
sort commands	*Sort the output based on the number of commands, highest number first. This option is only valid when used with the* **activity** *option.*
sort cputime	*Sort the output based on the amount of cpu time used, highest amount first. This option is only valid when used with the* **activity** *option.*
sort uci	*Sort the output based on the MUMPS UCI. This option is only valid when used with the* **activity** *option.*
sort elcount	*Sort the output based on the number of exclusive lock requests, highest number first. This option is only valid when used with the* **activity** *option.*
sort ewcount	*Sort the output based on the number of exclusive lock requests that had to be waited on, highest number first. This option is only valid when used with the* **activity** *option.*
sort ewtime	*Sort the output based on the amount of time waiting to get an exclusive lock, longest time first. This option is only valid when used with the* **activity** *option.*
sort lreads	*Sort the output based on the number of logical reads. This option is only valid when used with the* **activity** *option.*
sort preads	*Sort the output based on the number of physical reads. This option is only valid when used with the* **activity** *option.*

sort routine	*Sort the output based on the MUMPS routine name. This option is only valid when used with the* **activity** *option.*
sort runtime	*Sort the output base on how long a job has been running, oldest first. This option is only valid when used with the* **activity** *option.*
sort slcount	*Sort the output based on the number of shared lock requests, highest number first. This option is only valid when used with the* **activity** *option.*
sort swcount	*Sort the output based on the number of shared lock requests that had to be waited on, highest number first. This option is only valid when used with the* **activity** *option.*
sort swtime	*Sort the output based on the amount of time waiting to get a shared lock, highest number first. This option is only valid when used with the* **activity** *option.*
status	Show MUMPS and SYSTEM status information.
waiting	Show information with regards to waiting on lock queues.

Examples

```
show jobs
show jobs activity
show jobs activity sort cputime
show jobs activity sort lreads
show jobs dead
show jobs number 10 flags
show jobs 11 flags
show jobs 12 plt
show jobs pid 1234
show jobs status
show jobs flags pattern ZWAIT
show jobs waiting pattern BP
```

show locks

show locks [all] [counts] [job] [number N] [spinlocks] [type all,bilbufs,bp,general,jnlbufs,mumps,operator,volumes] [waittime]

Show the status of locks. The default is to show only general locks that have been used.

Table 9-12.

all	Show locks that have not been used.
counts	Show the number of times a lock has been locked exclusive, waited to lock exclusive, locked shared, and waited to lock shared.
job	Show locks that have a lock count not equal 0, and list jobs that have the lock on their Process Lock Table.
number N	*Where* **N** *is the lock number to show.*
spinlocks	Show spinlock counts or wait time.
type all	Show billbufs, bp, general, hashqueue, jnlbuf, mumps locks, operator and volume lock information.
type bilbufs	Show bil buffer locks only.
type bp	Show bp locks only.
type general	Show general locks only, this is the default.
type hashqueue	Show hash queue locks for buffer only.
type jnlbufs	Show journal buffer locks only.
type locks	Show hash queue locks only.
type mumps	Show MUMPS locks only.
type operator	Show operator locks only.
type volumes	Show volume locks only.
waittime	Show the wait times for a lock. This includes the spinlock pause time, wait for exclusive and wait for shared.

Examples

```
show locks
show locks jobs
show locks type general
show locks type bp number 25
show locks waittime
show locks all
```

show options

show options

Shows status of **mpinfo** internal options, page length, input file, output file, repeat last command, etc.

show swdstatus

show swdstatus

Shows the current status of the system watchdog program **mpswd** as follows:

Table 9-13.

ACTIVE	The **mpswd** process is running.
READY	The **mpswd** process is ready, initialization is complete.
COPYPF	*The* **mpswd** *process is coping the current ports file.*
READPF	*The* **mpswd** *process is reading the current ports file.*
RESTPF	*The* **mpswd** *process is restoring the previous ports file.*
SAVEPF	The **mpswd** process is saving the current ports file.
SLEEPING	*The* **mpswd** *process is sleeping until the next 10 second mark.*
TRACE1	*The* **mpswd** *process is logging trace level 1 information to a trace log file.*
TRACE2	*The* **mpswd** *process is logging trace level 2 information to a trace log file.*
TRACE3	*The* **mpswd** *process is logging trace level 3 information to a trace log file.*

show symbufs

show symbufs

Show the status of the symbol table copy buffers. Shows which job last used the buffer, the time and the use count.

show variables

show variables local [job NNN] [detail]

Table 9-14.

local	*Show a jobs local symbol table, must specify keyword* **job** *followed by a job number.*
job NNN	MUMPS job number, used with keyword job.

detail	Show lots of information about the local symbol table, addresses, etc.

Examples

```
show variables local job 10
show variables local job 1 detail
```

show waitqueues

show waitqueues [bp NNNN] [general GENERAL-SYSTEM-LOCK] [hashqueue NNNN]

Table 9-15.

bp NNNN	*Show the jobs waiting on the wait queue to lock bp* **NNNN** .
general GENERAL-SYSTEM-LOCK	Show the jobs waiting on the wait queue to lock a **GENERAL-SYSTEM-LOCK** .

bitmap bpoold bpoolg
bpoolh copybuf global
hogsys jnlupdate jnlwaiter
lockmgr mbilsmap msgbuf
quiesce setdev shrmem
svector symbuf1 wvector

Table 9-16.

hashqueue NNNN	*Show the jobs waiting on the wait queue to lock hash queue* **NNNN** .

Examples

```
show waitqueues bp 10
show waitqueues general svector
show waitqueues hashqueue 20
```

show wvectors

show wvectors [all] [jobs] [system] [used]

The default is to show all of the job and system wvectors.

Table 9-17.

all	Show all job or system wvectors.
jobs	Show only the job wvectors.
system	Show only the system wvectors.
used	Show only job or system wvectors that are used.

Examples

```
show wvectors
show wvectores job
show wvectors system used
```

Chapter 10. shutdown

Shutdown the M21 system.

shutdown database

shutdown database

Signals to **mpctl** to start a system shutdown sequence. This is useful when it is not possible to get into M21 to do **^SSD** .

WARNING..... Be certain that you really want to shut the database down, because once the shutdown starts it cannot be stopped.

Examples

```
shutdown database
```

suspend

suspend job

Suspend a running MUMPS job.

Chapter 11. suspend job

suspend job [number] NNN

This will cause a job to suspend executing and wait for the **resume job** command to be issued. THE JOB IS NOT RESUMED on quitting ***mpinfo*** but can be resumed subsequently by another ***mpinfo*** .

number **NNN** Job number to suspend.

Examples

```
suspend job number 10
suspend job 11
```

Chapter 12. unlock

Unlock a system resource that has been locked using the lock command.

unlock bp

unlock bp [number] NNNN

Unlock one of the buffer pointer locks. The lock has to have been locked with the **lock** command.

Examples

```
unlock bp number 10
unlock bp 11
```

unlock general

unlock general GENERAL-SYSTEM-LOCK

Unlock one of the **GENERAL-SYSTEM-LOCKS** . The lock has to have been locked with the **lock** command.

Examples

```
unlock general svector
```

unlock hashqueues

unlock hashqueue [number] NNNN

Unlock one of the hash queue locks. The lock has to have been locked with the **lock** command.

Examples

```
unlock hashqueue number 10
unlock hashqueue 11
```

unlock volumes

unlock volume [number] VV

Unlock one of the volume locks. The lock has to have been locked with the **lock** command.

Examples

```
unlock volume number 1
unlock volume 2
```

Chapter 13. wakeup

Force a job that is waiting on a lock to wake up.

wakeup job

wakeup job [number] NNN [clear]

Wake up a job waiting on a lock. This may cause error messages to be logged in the M21 log file if the job was not at the head of the wait queue for the lock it is waiting on.

clear If the job is at the head of the wait queue but it's flags are not zero, then clear them before waking it up.

Examples

```
wakeup job number 10
wakeup job 11
wakeup job 12 clear
```

wakeup pid

wakeup pid [number] NNNNN [clear]

Wake up a **pid** waiting on a lock. The **pid** is searched for first in the job wvectors and then in the system wvectors. This may cause error messages to be logged in the M21 log file if the job was not at the head of the wait queue for the lock it is waiting on.

clear If the job is at the head of the wait queue but it's flags are not zero, then clear them before waking it up.

Examples

```
wakeup pid number 12345
wakeup pid 23456
wakeup pid 34567 clear
```

Chapter 14. zap

zap job

Kill a running MUMPS job.

zap job

zap job [number] NNN

This is the same as killing a job from within the M21 environment using the MUMPS utilities.

Examples

```
zap job number 10
zap job 11
```