

## **M21 Release Notes**



## **M21 Release Notes**



## Table of Contents

1. Release Notes - M21 Version 1.06 - 02/08/2000.....	7
2. Release Notes - M21 Version 1.07 - 06/09/2000.....	9
3. Release Notes - M21 Version 1.07A - 08/09/2000 .....	11
4. Release Notes - M21 Version 1.07B - 14/09/2000 (HP).....	13
5. Release Notes - M21 Version 1.08 - 10/10/2000.....	15
6. Release Notes - M21 Version 1.09 - 23/11/2000 & 27/11/2000 .....	17
7. Release Notes - M21 Version 1.09A - 08/12/2000 .....	19
8. Release Notes - M21 Version 2.0 - 17/01/2001.....	21
9. Release Notes - M21 Version 2.01 - 24/01/2001.....	23
10. Release Notes - M21 Version 2.02 - 31/01/2001.....	25
11. Release Notes - M21 Version 2.03 - 13/03/2001.....	27
12. Release Notes - M21 Version 2.04 - 21/03/2001.....	29
13. Release Notes - M21 Version 2.05 - 14/05/2001.....	31
14. Release Notes - M21 Version 2.06 - 16/07/2001 & 13/08/2001 .....	33
15. Release Notes - M21 Version 2.07 - 17/10/2001.....	35
16. Release Notes - M21 Version 2.08 - 15/04/2002.....	37
17. Release Notes - M21 Version 2.09 - 13/05/2002.....	41
18. Release Notes - M21 Version 2.10 - xx/xx/2002.....	43



## Chapter 1. Release Notes - M21 Version 1.06 - 02/08/2000

All M21 utilities now use the command/shell history facility, including the debugging software ( `%MuDbg*` ). The command/shell history problem when changing direction has been fixed.

Two additional flags are available for the `^g` option of `^%MuGlo` . These are `/Quote` for quotes to be displayed around the data (as per the MSM global lister) and `/Doublequote` for quotes to be displayed around the data and any quotes within the data to be doubled up (more strictly correct then `/Quote` ). The `^%GL` 'legacy program name' automatically takes the `/Quote` flag.

A global search (via `^%G` , `^%MuGlo` or `^%GSE` ) now searches global reference details as well as data for the search string (as per the MSM utility).

Changes have been made to the `^%MuGlo` and `^%MuRou` outputs so that, when help has been toggled off, the prompt is shortened to Global(s) or Routine(s) and fewer blank lines are displayed. When these utilities are called by 'legacy program names', such as `^%GL` , `^%FL` and `^%RSE` , help is automatically toggled off on entry to `^%MuGlo` / `^%MuRou` . `^%G` and `^%R` are not treated as 'legacy program names' but as shortcuts to `^%MuGlo` and `^%MuRou` , therefore `D ^%GL` under M21 is the closest equivalent to `D ^%G` under MSM.

Global and Routine saves now default to the 'no-encode' format, which is compatible with the MSM Global and Routine restore. In line with this the `/Noencode` flag no longer exists and there is now a `/Encode` flag which encodes non-printable characters and the { and } characters to the {`nnn`} format during the save. The M21 Global and Routine restore utilities check whether the saved format is 'M21 encoded' or not, and restores appropriately.

Error trapping has been changed within the `^%MuGlo` utility to give more details when an error occurs whilst traversing a global to find nodes matching the search criteria. The values of `%MuGlo` , `%MuNC` , `%MuSub` , `%MuDF` , `%MuData` and current subscript " F "rom," T "o & " C "ondition parameters are given. Therefore, if `%MuData` or `@%MuGlo` values are used in the search criteria at a level where `%MuDF#10=0` an `<UNDEF>` will occur and the values described above will be displayed, indicating how far the transversal had progressed. Where such an `<UNDEF>` error might be expected a `$Get(%MuData)` function can be used to treat non-existent data as null, therefore avoiding the `<UNDEF>` error.

Error trapping has been improved in the `^%MuRou` utility, so that, when appropriate, warnings are given when error situations are encountered (e.g. `<PROT>` , `<ISYNT>` or `<CLOBR>` ) rather than the utility aborting.

Problems with `/L` and `/R` within the debugging software not invoking the correct response have now been fixed.

The originating device is now correctly set up for logins via `^%MuLogon` .

The programmer shell has been modified to give trace back details and, where appropriate, leave the routine causing the error in the programmer shell. Note that this information is given by using `$ETRAP` - existing error trapping routines should set `$ECODE` to null to ensure that the correct error information is displayed at the programmer shell.

A system status list can now give the user ID of the process if requested. Note that obtaining this information from the operating system may take a couple of seconds. If the user ID is not required there is **^%SSS** (short system status) which does not process the user ID.

**\$ZDATE** - uninitialised variable mode causing random **<SYNTAX>** and memory errors when the second parameter is not supplied to the **\$ZDATE** function.

Numeric global subscripts that are greater than 16 digits in length are displayed as numeric rather than string values, which causes problems with **\$ZREFERENCE** and **\$QUERY**.

Display not updating correctly when the UCI wide journaling flag is changed in **UCIMGR** option 4.

Added **\$ZTEXP** function to raise e to the power of the supplied parameter - note must be spelled in full to differentiate from **\$ZTEXT**.

Added **\$ZTL[OG]** function to compute the natural logarithm of the supplied parameter.

Added read access **\$ZINFO** for **zsaveopt** in **pvector**.

Problem with memory error when **\$ETRAP** is set and a routine is called either by **DO** or **\$\$** and the parameters are in error causing a **<DPARM>** error prior to routine execution. This was caused by the `long jmp` environment not being set up prior to the error occurring.



## Chapter 2. Release Notes - M21 Version 1.07 - 06/09/2000

### System Status Utilities

The elapsed time display has been changed so that the time is converted to elapsed days, if appropriate. Also, there are now several formats available. In most output formats the device columns have been combined into one column, with **o** to represent originating device, **c** to represent the current device and **p** to represent principal device. The system status options are as follows:

All the available information is shown via **D ^%SS** or **D ^%MuStat** or **\$\$Status^%MuStat()**. This information includes details of 'kept background jobs' - **KZJOB**s. (As each M21 job has its own UNIX process, then whenever a background job is started a UNIX process is also required. If there is a UNIX process already waiting to be used then the processing time to start the background job is reduced.)

- **D ^%SSS** or **\$\$Status^%MuStat(,,1)** gives a shorter output as it does not display the **KZJOB**s
- **D ^%SSC** or **\$\$Status^%MuStat(,"compact",1)** gives a more compact output and doesn't display the Hang time or the elapsed time.
- **D ^%SSQ** or **\$\$Status^%MuStat(,"quick",1)** gives a quicker output as it does not interrogate the UNIX environment for the username details.
- **D ^%SSO** gives the original output format with no username and 4 columns for Device details.

The status display of the **^KILLJOB** utility uses the **^%SS** format, so **KZJOB** details are visible.

The status display of **JOBEXAM** uses the **^%SSS** format as it would not be appropriate to examine the **KZJOB**s.

The programmer shell has been modified so that it now deals with a **<LOMEM>** error.

The programmer shell has been modified to give a **<SYNTAX>** error when **<TAB>** is used incorrectly. For example ' **s x=<TAB>** ' will now give an error. Previously it treated the **<TAB>** to mean it was a line of code with a line label of **s**.

The **^%MuDev** utility has been modified so that it uses the first unopened host file device for the **M** process - remember M21 host file devices are private to the **M** process. Previously it always used the first host file device (8000) irrespective of whether it had previously been opened.

All **^%Mu\*** utilities that opened a host file without using **^%MuDev** have now been changed to use **^%MuDev**.

**READ \*** -causing random **<MXMEM>** errors when used with socket or HFS devices.

The permissions on creation of the sub-directories of the database directory - **%MuErr**, **%MuExt** and **%MuRvi** have been changed from 0775 to 0755.

The maximum number of open devices has been changed from 25 to 50.

**\$ETRAP** is no longer stacked on **NEW \$ETRAP** or **SET \$ZTRAP=value** , if **\$ETRAP** has already been stacked at the current execution level.

The owning device ( `apv->hdzo` ) is now set to the same as the principle device ( `apv->pridev` ) when a partition is created in non-tied mode.

Exceeding the maximum number of execution levels with an **XECUTE** caused either a **<STAP>** error or a memory access error. This has now been changed to give a **<LEVEL>** error as for **DO** - however it still gives a **<SYNTAX>** or **<CMMND>** error when running under `%MuPGS` .

When the M database filled up the error returned was **<SYNTAX>** rather than **<DKFUL>** , this has now been corrected.

**\$ZUCI** sometimes returned information for the wrong UCI. For example, if UCI 10 was **AAA** and UCI 100 was **BBB** , then **SET x="100" S \$E(x,3)="" W \$ZUCI(x)** returned the information for UCI **BBB** rather than that for UCI **AAA** . This has now been corrected.

**MERGE** from a global to a local sometimes inserted nodes into the local variable at the wrong point causing data traversal functions such as **\$ORDER** to loop.

The mnemonic **\WAIT(0)** on a TCP/IP device waited for ever rather than returning immediately. This behaviour has been corrected.

## **Chapter 3. Release Notes - M21 Version 1.07A - 08/09/2000**

Under certain circumstances a subscripted variable that was passed by reference to an extrinsic function could lose its subscripts when another subscripted value of the variable was set in the extrinsic function or a subroutine called by it.



## Chapter 4. Release Notes - M21 Version 1.07B - 14/09/2000 (HP)

The fix to **\$ZUCI** in 1.07 sometimes caused a **<MXMEM>** error to occur when using the **\$ZUCI** function.

When an error occurred at a level where **\$ZTRAP** was set and then a return was made to a level where **\$ETRAP** was set without clearing **\$ECODE**, the **\$ETRAP** code was not executed. This was caused by a flag to indicate that an error had occurred only being set when **\$ETRAP** was non-null.

A **<SYNTAX>** error rather than a **<LEVEL>** error occurred on repeated executes within the programmer shell, this was caused by memory overflow. This behaviour has been corrected by changing the **\$ETRAP** error trap routine to `malloc` its own memory.

A memory error rather than an **<UNDEF>** error would sometimes occur on the HP-UX platform, this was caused by corruption when compiling the **\$ETRAP** code and was fixed by the `malloc` of memory within the error trap mentioned above.



## Chapter 5. Release Notes - M21 Version 1.08 - 10/10/2000

Programmer shell trace back stack is now implemented except for execute at the bottom level.

M21 workstation VCM client interface now correctly reports when a routine is not transferred to the server due to the first lines being the same.

M21 workstation client interface now correctly handles routines with no sources.

M21 workstation client interface now correctly handles the external telnet option when `^Mu("WCI","Telnet command")` is set.

If the system time was set back, either for daylight savings time or because of an error in the time, then if there were unwritten modified buffers at the time the change was made the system would fail to perform the next quiesce and after several retries, the system would be shut down.

If a kept background job was killed when it was in an idle state, then the internal background job counters were not updated correctly and problems could occur when starting future background jobs.

The change in 1.07B to `malloc` memory within the error trap and reset the stack pointers caused spurious `<ESTAP>` errors within the programmer shell error trap on the HP-UX and AIX platforms, this has now been corrected.

Passing a variable by reference to a function using `$GET` could cause `<MXMEM>` errors or incorrect results when testing against the variable dependent on platform.





## Chapter 6. Release Notes - M21 Version 1.09 - 23/11/2000 & 27/11/2000

When using the **MERGE** command to copy from a local variable to a global variable memory could become corrupted causing a <MXMEM> error.

When using the **MERGE** command to copy from a global variable to a local variable or from a global variable to a global variable, if the source global included negative numeric subscripts, then the variable structure was not replicated properly in the target local or global.

Unusual exit from `mpmse` messages could occur in the log due to a premature exit from `part()` caused by turning off programmer mode within the application code when the **HALT** had not yet been processed in the input file, the internal code has been changed to carry on processing input whilst the input file flag is set.

M21 has now been ported to Linux version 6.2.

If the **ZTRAP** command were specified either in full or using an abbreviation greater than two characters, then a <SYNTAX> error would occur.



## Chapter 7. Release Notes - M21 Version 1.09A - 08/12/2000

When opening a TCP/IP listening socket on Linux, if an IP address of null was specified then the socket would not be established and an error of `M98, Resource unavailable` would result in `$DEVICE`. This was caused by an internal function to return IP address not returning 0 when a null hostname was supplied. This problem only affected the Linux platform.

The debug utilities did not work on the Alpha platform due to the difference in how data types are represented by the underlying architecture.



## **Chapter 8. Release Notes - M21 Version 2.0 - 17/01/2001**

The debug utilities did not work on the Alpha platform due to the difference in how data types are represented by the underlying architecture.

**\$ZINFO** has now been modified to allow the setting of string values into internal structures.

There was a problem on 64 bit platforms caused by using offsets into the command table rather than array indexing. This could cause a variety of memory access errors.

Memory allocated for debug purposes was unnecessarily zeroed. This code assumed that the size of a pointer was four bytes, which caused problems on machines with a different pointer size, such as the Alpha.

Various unexpected errors could occur whilst debugging due to the overrunning of a memory buffer used for storing routine lines and execution strings. This problem has now been corrected.



## Chapter 9. Release Notes - M21 Version 2.01 - 24/01/2001

There was a problem when multiple jobs used inter job communication (IJC) and the number of pipes available had been set in the configuration file to be more than the default of 10. This problem would manifest itself by all **LOCK** and **ZALLOCATE** commands across all jobs failing with a <MXMEM> error. In addition the system could freeze up due to the master shared memory and lock table system locks not being released by the first job to fail to lock. This was a problem with more than 10 IJC channels, where the shared memory area for the IJC device descriptor blocks (DDB) was set up for the number of IJC devices (10) rather than number of possible IJC channels. When IJC channels above the tenth were opened the DDBs first overwrote the Magnetic Tape DDB area and then the lock table, causing the <MXMEM> errors.

The minimum, default and maximum values for inter job communication channels have been changed to 16, 32 and 2048 respectively.





## Chapter 10. Release Notes - M21 Version 2.02 - 31/01/2001

This release includes a port to the 64 bit DEC-ALPHA platform running OSF1 V4, in addition to the existing ports.

Changes have been made to the debugger to correctly honour naked indicators and to stop random errors which occurred in certain circumstances, such as when routines lines were greater than 255 characters

**^%SSU** is a new System Status utility that just lists the jobs running in the same UCI as the system status.

Changes have been made to the routine compare software to ensure that lines are correctly shown.

The **^%E** editor has been modified so that if there is a blank line in the code, the coding after the blank line is not lost.

Changes have been made to the API and Fast Global Copy software so that the listening socket is closed if it cannot be detached. Without this change, further connections fail if there are UNIX resource problems; with the change connections continue, albeit less efficiently.

The M21 error trapping software has been modified so that the variables are now trapped via the default error trap, in the situation where a background job, without its own error trapping, crashes.

*Where appropriate, all routine, source and global utilities now return to the selection prompt after processing. Previously there were inconsistencies throughout the utilities. When the selection is not kept (i.e. not **^%R**, **^%G** or **^%S**) then entry of **^k** will retain the previously selected list. After a search the search prompt is now repeated.*

The pipe naming convention of database name followed by a three digit number did not allow for the configuration of more than 1000 inter job communication devices. The name has now been changed to allow four digits.

The changes to **\$ZINFO** to allow the command to work on the 64 bit alpha platform caused a **<MXMEM>** error on normal byte ordered machines (HP-UX and AIX) when a numeric value or pointer greater than 2147483647 was returned. This problem has now been corrected.



## Chapter 11. Release Notes - M21 Version 2.03 - 13/03/2001

There was a problem with routine indirection when this was followed by another command on the same line where the compiler did not correctly position the compile pointer and an **<INDER>** error resulted.

**\$PIECE** with a large index could cause a **<MXMEM>** error with unpredictable consequences for the running M21 system.

**\$EXTRACT** with a large index could cause a **<MXMEM>** error with unpredictable consequences for the running M21 system.

If M21 was running using a user id other than the default ( `m21` ), then attempting a backup would fail due to the master backup process not passing the user id on to the slave backup process.

The `runbackup` script did not allow for the specification of a user id and so automatically assumed that M21 was running using the `m21` user id. It is now possible to specify a `-uUSER` parameter to the script.

As a default, starting the cross system request server automatically checked the network for DDP messages. This would cause M21 to M21 communications to fail where DDP was either not implemented or not configured correctly. This was a problem which manifested itself mainly on IBM RS/6000 systems. Networking has now been changed to NOT start DDP as a default.

A **\$NAME** expression using a naked reference, where **\$ZREFERENCE** was updated during evaluation of the **\$NAME** value, would in certain circumstances return an incorrect result string.

**<CMMND>** or **<SYNTAX>** errors occurred when some commands were partially spelled, that is, specified using neither the minimum or maximum number of letters in the command name. e.g **WR** for **WRITE** .

The string returned by **\$ZVERSION** has now been changed to start with **M21-UNIX** rather than **M21** .

The behaviour of journaling routine changes and deletions was incorrect and did not honour the partition journaling flags - this has now been corrected.

The length of the internal buffer used by the M level debugger has been increased to 1024 bytes.



## Chapter 12. Release Notes - M21 Version 2.04 - 21/03/2001

When running the kill job utility ( **^%MuKJob** , **^killjob** , **^%killjob** , **^KILLJOB** , **^%KILLJOB** , **^resjob** , **^%resjob** , **^RESJOB** or **^%RESJOB** ) or the job exam utility ( **^%MuEJob** , **^jobexam** , **^%jobexam** , **^JOBEXAM** or **^%JOBEXAM** ) there is now an option of **Ucistatus** to show the jobs in the current UCI

There is a new utility of **^%SST** that gives a system status showing the busiest jobs first. There is also **^%SSTU** , which gives a system status of all jobs for the current UCI, showing the busiest jobs first.

There is a new vector of **INT^%T** which returns **%TIM** and **%TIM1** in line with the MSM utility.

There is a new vector of **INT^%D** which returns **%DAT** and **%DAT1** in line with the MSM utility.

When using the **^Delete / Noask** option of **^%G** or **^%MuGlo** with only whole globals specified, if the global **^MuTmp** has been selected by the selection criteria, the deletion of **^MuTmp** will be done last. Previously the deletion of the **^MuTmp** global in its correct alpha sequence meant that globals that followed **^MuTmp** were not deleted.

Although the **^%GDEL** vector still works as before there is now a new **NOASK^%GDEL** vector that works as the **^Delete / Noask** option described above.

The default error logging routine (called either because no error trap has been set, or because the error trap is set to **^%ET** ) now checks how many errors have been logged for the UCI that day. If 200 errors have already been logged then the variables at the time of the error will not be recorded. Note the value of 200 is the default value. It can be changed by setting up **^Mu("%MuErr","MaxPerDay")** to the number of errors where you want to store the value of the variables.

If the Lock Table changes rapidly it may not be possible for a Lock Table display to show all the locks at a given point. To avoid the utility crashing an error trap has been added and the following message is given - "Aborted - Lock table is changing too rapidly - Try again"

There is now help for the 'Control Device for another job' option

**^Mu("%MuLogon",\$j)** nodes are now written at login.

**^%MuFGC** client (Fast global copy over a TCP/IP socket) now works for ISM and DSM in addition to M21 and MSM.

The problems with locktable on the DEC port have been resolved.

There is now a vector of **^%L** to run the locktable.

There is now a vector of **^%LU** that displays the locks for the current UCI.

There is now an error trap in the System Status Utility.

After freeing or assigning a block in a Map block, the Database Fix utility now asks if you want to update the free block counters.

Further modifications have been made to the debugger to stop memory access errors which occurred in certain circumstances when debugging was disabled.

**SET \$EXTRACT(var,m,n)** when **n** is less than **m** incorrectly gave a <FUNCT> error, rather than doing nothing.

Multiple argument **SET** s which include **\$EXTRACT** or **\$PIECE** in the argument list would result in incorrect values being set into some of the variables.

Under certain circumstances, inserting a routine line containing parameter names longer than eight characters could corrupt the routine line causing <ISYNT> errors.

## Chapter 13. Release Notes - M21 Version 2.05 - 14/05/2001

^%MuFGC client (Fast global copy over a TCP/IP socket) now works for ISM under VMS and for Cache.

System status displayed an incorrect value for "Elapsed" when the system date and time had been put back to before the start of the job. It now returns a value of zero under these circumstances.

There are now vectors of ^%J and ^%JOB to run the ^JOBEXAM utility.

On an IBM AIX system the ZWRITE command now works correctly.

The journal utilities now correctly skip over backup entries in the journal file.

Internal block formatting functions used by the journal and database utilities now handle negative numeric subscripts correctly.

When tracing was turned on in a background job to another system device, then closing the device would cause an error and, if the background job was a kept background job, it would terminate. The error could not be trapped or processed.

On the Alpha platform, inter-job communication devices would not work correctly following a system restart. The only cure for this was to remove the pipes directory and its contents prior to starting the M21 system.

When evaluating a \$ZINFO(11... call the stack pointer was not updated correctly, this could cause problems when the \$ZINFO(11... was used as part of an expression. These problems included unexpected errors or corruption of variables used as part of the expression.

On creation of a background job, the journal partition and do not journal partition flags were not copied correctly.

The function \$ZINFO(1,"pvector","hdzeâ) would return an incorrect value when the system date and time had been put back to before the start of the job. The function now returns a value of zero under these circumstances.

The function \$ZINFO(1,"pvector","hdzeâ) would always return a redundant second piece of 0 delimited by colon. This has now been removed.

When calling a routine, sub-routine or function that changes UCI, then, on return, the wrong routine could be loaded and executed. This would only happen if the routine name of the calling routine and the called routine matched for the length of the calling routine and the called routine name was longer.

When starting a background job using a kept ZJOB the start date and time was not reset. This caused the elapsed time display in system status to show incorrect information for that job.





## Chapter 14. Release Notes - M21 Version 2.06 - 16/07/2001 & 13/08/2001

Entry of **ZQ** at the programmer shell prompt now gives an error and processing continues; previously **ZQ** at the programmer shell prompt resulted in the process erroring and exiting.

There is a new utility of **^%MuTn** that allows telnet access to another system, with access back to the original system.

The delete option of **^%MuGlo** has been corrected so that nodes that don't exist are not presented with a Delete ? <N> prompt.

A ' **nice server** ' has been written for use on systems where there are jobs that write huge amounts of data in a short time (e.g. the production of the metrics on the sprite system). When the ' **nice server** ' is running then any jobs identified via the existence of a **ZAllocate** of **%Mu("Nice",\$J)** will be suspended during a quiesce (i.e. the process that occurs every 15 minutes that records all block changes since the previous quiesce). This will ensure that no processes/users are locked out during the quiesce, which occurs when data is being put into disk buffers faster than the data can be written out to disk. The 'nice server' will have more effect on systems that have fast processor speeds with relatively slow disk access speeds and will have little effect on systems that have slower processors with relatively fast disk access.

**^ %MuFGC** (Fast global copy via a TCP/IP socket) has been modified to **ZALLOCATE %Mu("Nice",\$J)** so that the data writes are suspended for the quiesce period.

The **^ %E / ^%MuRvi** editor link to vi has been enhanced to check for **VC/m** controlled routines.

The programmer shell error handling and traceback stack display have been enhanced.

Access to processor ( **cputype** ) and operating system vendor ( **vendor** ) are now available using **\$ZINFO(9...**

Access to the input ( **IBUFSIZE** ) and output ( **OBUFSIZE** ) terminal buffer sizes are now available using **\$ZINFO(4...**

The default terminal input buffer size has been changed from 256 bytes to 512 bytes.

Errors occurring when opening a TCP/IP device for socket communications were not correctly reported in **\$DEVICE** . This could sometimes lead to the incorrect assumption that the open has succeeded.

The **\$STACK** function returned incorrect results for **\$STACK(\$STACK)** and **\$STACK(\$STACK,"MCODE")** following an error with **\$ETRAP** set. **\$STACK(\$STACK)** returned **DO** or **EXECUTE** rather than the error which had occurred and **\$STACK(\$STACK,"MCODE")** returned the **\$ETRAP** value rather than the code which was executing at the time of the error.

**dbinit** has been changed to set up a **%MuTrc** sub-directory and to correct directory names and permissions.



## Chapter 15. Release Notes - M21 Version 2.07 - 17/10/2001

DDP is now available for the IBM AIX platform.

Two new **ZMTRACE** commands have been added - 3001 to suspend an M job and 3000 to resume a suspended M job. . **BEWARE suspending your own job may suspend the entire M21 system.**

A tuning parameter has been added to optimise the use of buffer cache for a given M21 system. **GET\_BP\_SLEEP** is a system wide dynamic configuration parameter that can be specified in the configuration file. This parameter specifies the time in 1/100 of a second to sleep between attempts to lock a buffer cache block that is in use. The default value is 50 (1/2 second) and values can be specified between 1 and 50. The default value is the same as that implicit on previous versions of M21 allowing for complete backwards compatibility. A special value of zero has also been implemented. This adds each job waiting for a buffer that is in use to a first in first out queue rather than trying to lock the buffer repeatedly in a loop. **WARNING, injudicious use of this parameter can seriously degrade the performance of an M21 system.**

A tuning parameter has been added to optimise the use of M locks for a given M21 system. **LOCK\_RETRY** is a system wide dynamic configuration parameter that can be specified in the configuration file. This parameter specifies the time in 1/1000 of a second to sleep between attempts to lock an M global that is already locked. The default value is 500 (1/2 second) and values can be specified between 1 and 3000. The default value is the same as that implicit on previous versions of M21 allowing for complete backwards compatibility. **WARNING, injudicious use of this parameter can seriously degrade the performance of an M21 system.**

The output from the M level trace ( **ZM 1** ) has been modified to add a timing value to each line. This is the value of the system clock tick at the time of the output.

Access to the value of the current system clock tick is now available using **\$ZINFO(9,"tick")** .

A tuning parameter has been added to force a process to sleep after every 100 commands executed. This value can be accessed using **\$ZINFO(1,"pvector","p\_nice\_sleep")** and set using **\$ZINFO(7,"pvector","p\_nice\_sleep",value)** . The value is specified in milliseconds and the default is zero allowing compatibility with previous versions of M21. **WARNING, injudicious use of this parameter can seriously degrade the performance of an M21 system.**



## Chapter 16. Release Notes - M21 Version 2.08 - 15/04/2002

Performing a **\$QUERY** , **\$ZORDER** or **\$ZNEXT** on a translated global that ended up going across systems, incorrectly returned the environment specification (extended syntax) as part of the global reference. The Naked indicator was, however updated correctly. This problem has now been corrected.

The cross-system request server ( `mprequestsrv` ) now cleans up properly after itself if it is terminated with a **SIGTERM** (15) signal.

On 64 bit platforms such as the Compaq/DEC Alpha, internal pointers are stored as 8 byte values rather than 4 bytes on 32 bit platforms. **\$ZINFO** , whilst capable of examining these pointers using **\$ZINFO(1,..** , was not capable of setting these values using **\$ZINFO(7,..** . The value to set the pointer to was internally truncated to 32 bits and the pointer set incorrectly. The result of this was that various utilities, including the debugger and the translation/replication utilities would not work correctly on 64 bit platforms. **\$ZINFO** has now been modified to allow the setting of 64 bit values into pointers on 64 bit platforms.

A new dynamic configuration file parameter **VIEW\_NOT\_IMPLEMENTED** has been added. **VIEW\_NOT\_IMPLEMENTED** should be specified with a value of **YES** in order that usage of the **VIEW** command or **\$VIEW** function will produce a **<NOIMP>** error. The default value of this parameter, if not specified, is **NO** . This capability is especially useful when porting new applications to M21. This will force errors when a **VIEW** command or **\$VIEW** function is encountered rather than these going unnoticed and causing random errors.

A new static configuration file parameter **REQUESTSRVDEBUG** has been added. **REQUESTSRVDEBUG** should be specified with a value of **YES** in order that the cross system request server will start up with debugging enabled . The default value of this parameter, if not specified, is **NO** . The request server will write debug information to a file in `/tmp` called `rs.xxxx` where `xxxx` is the process id of the request server.

A new static configuration file parameter **STARTDDP** has been added. **STARTDDP** should be specified with a value of **YES** in order that the cross system request server will start up with processing of DDP messages enabled . The default value of this parameter, if not specified, is **NO** .

A new static configuration file parameter **NETWORK\_CARD** has been added. **NETWORK\_CARD** should be specified with a value of the device identifier of a network card to use for DDP. The default value of this parameter, if not specified, is null. The default value will mean that the request server will choose the configured network card with the lowest number, ignoring the loopback device.

A new dynamic configuration file parameter **VIEW\_PROTECT** has been added. **VIEW\_PROTECT** should be specified with a value of **YES** in order that usage of the **VIEW** is restricted to the Manager UCI and library (%) routines. The default value of this parameter, if not specified, is **NO** .

When an **OPEN** was carried out on a device which was already **OPEN** 'ed by the same job, then, if a timeout was specified but no other parameters, **\$TEST** was not updated to indicate success.

The M(UMPS) sorts after operator **]]** has been implemented.

User based Licensing has been implemented.

When a **MERGE** command was performed with the flag to treat undefined globals as null set then global nodes would be set in the target global even though they were not present in the source global. This behaviour is illustrated by the following code:

```
; Set the flag to treat Undefined globals as NULL
W $ZINFO(8,"pvector","UNDEF_GLOBAL",1)
KILL ^X,^Y
MERGE ^Y("TEST","WK")=^X("TEST","WK")
```

Run this and **^Y("TEST","WK")** gets set to null even though **^X** is not defined.

There is a case for saying that this is correct behaviour in these circumstances since a **MERGE** command can always be equated to a series of **SET** commands and, with the flag to treat undefined global variables as NULL, a **SET X=^X("TEST","WK")** would (correctly) set **X** to NULL. Then by extrapolation the **MERGE** should set **^Y("TEST","WK")** to NULL in the above example (which it did). However, this behaviour was not the same as that on other versions of M(UMPS) with the same flag set, so this behaviour has been corrected to not set **^Y("TEST","WK")** in the above example.

The `runbackup` script used for performing an unattended backup of the M(UMPS) database had an error in the parameters which would result in journal backups not completing.

Domains can now be specified and used with host spool devices.

If domain or device translation parameters were specified on a device **OPEN** command and a timeout was not present, then these parameters would be confused and the **OPEN** would fail with an error. This has been corrected.

The loading of mnemonic namespaces at IPL of M21 did not work correctly due to an error in one of the IPL utilities. This error has been corrected.

A general socket server utility **^%MuSrv** has been introduced which can be started at system IPL time, listen on a specified port and despatch user routines specified in configuration parameters. Details of this generalised mechanism are as follows:

- The general socket listener process **^%MuSrv** provides a mechanism for listening on a given port and dispatching a user routine when a connection is made to that port. On entry to the user routine the current device for the routine will be set to the first socket device - 9050.
- Each of the listeners is identified by a unique `index`. The following global nodes are used to set parameters for each socket listener process that is to be started.
- **^Mu("Srv",index,"Maximum log entries")="No"**  
Specifies the maximum number of log entries to keep if logging is enabled. If this parameter is not specified the default is 500.
- **^Mu("Srv",index,"Maximum queue size")="No"**

Specifies the maximum queue depth for incoming connections. If this parameter is not specified the default queue depth is 5.

- **^Mu("Srv",index,"Port")="PortNo"**

Specifies the port number for the server to listen on. If this parameter is not specified the default port number is 2001.

- **^Mu("Srv",index,"Routine")="RoutineRef"**

Specifies the user routine to run when a connection is made to the socket server. This parameter must be specified.

- **^Mu("Srv",index,"UCI")="UCIName"**

Specifies the three-character UCI name in which to run the specified user routine. The routine must exist in the UCI specified. If this parameter is not specified the default UCI is MGR.

- The listeners specified in the global **^Mu** will start automatically at system IPL time if the following parameter is specified in the configuration file:

**IPL\_StartMServers=YES**

- When started at IPL time logging is not enabled for a socket server process.

In order to stop a socket listener process identified by `index` set the following global node:

**^Mu("Srv",index,"Stop")=""**

In order to start an individual socket listener process identified by `index` :

**JOB Server^%MuSrv(index,log)**

Where `log` is 0 (zero) for no logging and 1 to log diagnostic messages in subnodes of the global **^Mu("Srv",index,"Server log")**

A previous fix to `JobWait^%MuUtil` introduced a problem with calling external commands on Linux systems. This has been corrected.





## Chapter 17. Release Notes - M21 Version 2.09 - 13/05/2002

Support for the ANSI standard error codes for domain and mnemonic related errors have been added

Some problems have been corrected in the licensing support algorithms.

Specification of the shortened form of the intrinsic special variable name ( `svn` ) `$D` did not default correctly to `$DEVICE` but produced a `<FUNC>` error instead. This has now been corrected to default to the `svn $DEVICE` .

The intrinsic special variable name ( `svn` ) `$DEVICE` should, according to the ANSI standard, return a value of 0 (zero) for success and a non-zero value for failure. In previous versions `$DEVICE` incorrectly returned an empty string rather than zero for success. This has now been corrected to reflect the ANSI standard behaviour.

Running the external utilities `mpctl` , `dbbkrs` , `licence` , `dbinit` , `dbuci` , `dbverify` , or `mpinfo` without the shell environment variable `$BASE` set would cause a segmentation fault to occur and the utility would not run. This has been corrected so that the value of `$BASE` , if not present, is obtained from the value of `$BASE` that was available when the M(UMPS) system was first started.



## Chapter 18. Release Notes - M21 Version 2.10 - xx/xx/2002

There was a problem with the sorts after operator `||` such that, when two variables containing integer values were specified as operands to the operator, then, under certain circumstances, an incorrect truth value could be returned. This problem has now been corrected.

Formerly, the working directory for background jobs started in M21 defaulted to `/tmp`. This has now been changed to default to the value of the **\$BASE** shell environment variable at the time of starting the M21 system.

When the **MERGE** command was used to copy a global variable to a global variable or a local variable, if any numeric subscripts longer than 17 digits were present in the source global, then the first 17 digits of the subscript would be copied to the destination global and any further digits would be replaced by zeros. This problem did not occur when **MERGE** ing a local variable to a global variable, or a local variable to a local variable. This problem has now been corrected.

There was a memory allocation problem associated with the Windows version of the M21 API. When the functions to login in or out of the M21 system were called, memory was allocated that was then not freed until the M21 object was destroyed by the client application. This behaviour would result in an application which used the API crashing when it had exhausted virtual memory on the Windows system, if the same M21 API object was used to log in and out of the M21 system repeatedly. This problem has now been fixed.

