

HP e3000 MPE/iX Sort and Millicode FAQs

Q1: When did HP learn about the Sort and millicode issues?

A1: *HP just recently became aware of these issues. HP has worked aggressively and quickly to resolve these issues.*

Q2: Can these situations occur on earlier MPE/iX releases like 5.5 or 6.0?

A2: *No, since the behavior requires the use of large files and large file support was introduced in release 6.5 on March 30, 2000.*

Q3: When did the exposure to these issues start?

A3: *The exposure began with release 6.5 and the use of large files.*

Q4: What is the status of the patches?

A4: *Both patches MPENX11 and MILNX10 are in General Release (GR) status and available on the ITRC or through HP Support.*

Q5: Can the patches be staged?

A5: *Patch MPENX11 can be staged with Patch/iX however MILNX10 cannot be. If patch MPENX11 is installed by itself it may be staged and eventually installed by rebooting the system. However, installing MPENX11 with MILNX10 will force PatchiX to create a tape.*

If you install MILNX10 by itself with Patch/iX you will be prompted to create a tape however it WILL NOT require a reboot to be officially installed as Patch/iX simply replaces MILLI.LIB.SYS and then STREAMs an installation job (IHFNX10[A|B|C]) to create the files /lib/libmilli.a and /lib/libmilli.sl with the appropriate file access permission settings. It is important that the installation job be streamed to ensure that applications developed via POSIX interfaces like c89 will include the new millicode.

Q6: Do these patches resolve the issues for MPE OS?

A6: *Patches MPENX11 and MILNX10 address these issues for all MPE/iX FOS and Subsystem software. Programs or libraries that use file system intrinsics such as FREAD or FREADDIR or language I/O such as the PASCAL/iX or C/iX read() functions are corrected when MPENX11 has been installed.*

Q7: Why does scanning object code for references to \$\$lr_unk_unk_long (or \$\$lr_na_unk_long) have little value?

A7: *Hundreds, if not thousands of programs and libraries may reference these symbols without causing any problems. There are a number of factors that must be considered in determining whether an executable may be susceptible to this issue:*

- *Does the executable call HPFOPEN?*
- *Do any calls to HPFOPEN pass item 87 for large file mapped access?*

- *Does the executable also use the PASCAL predefine move_fast passing it a large file pointer as the source address and request the transfer of 6 or fewer bytes.*
- *Or does the executable use PASCAL or C long pointer constructs to move 6 or fewer bytes from a large file?*

The HP recommended approach is for customers with in-house applications written in Pascal/iX or C/iX to review their source code to see if they meet all the criteria listed above. If all the criteria are met, then the application should be rebuilt with the MILNX10 patch installed on the system. As already mentioned those applications which use file system intrinsics or language I/O functions are corrected with the application of MPENX11.

Q8: *Should I install just MPENX11 if I'm not recompiling any applications?*

A8: *HP recommends that you install both patches so no new situation could be introduced.*