

Inter-Office Memorandum

To	AlphaMesa Users	Date	May 19, 1980
From	John Wick	Location	Palo Alto
Subject	Mesa 6.0 Pre Alpha Test Change Summary	Organization	SDD/SS/Mesa

XEROX

Filed on: [Igor]<AlphaMesa>PreAlpha60.bravo (and .press)

DRAFT

This memo outlines changes made in Mesa since the last release (Mesa 5.0, April 9, 1979); it summarizes important operational differences which you must know in order to use Mesa 6.0. A more complete change summary and the usual update memos are forthcoming.

References

In addition to this memo, preliminary versions of the Mesa 6.0 System Update, XMesa Update, and RunMesa Update are available on [Igor]<AlphaMesa>Doc> as System60.press.

General

On large programs, performance of the Mesa 6.0 Compiler and Debugger on 64K Altos is considerably worse than in Mesa 5.0; *an extended memory machine with a minimum of 128K is recommended.*

Alto Operating System version 17 and Executive version 10 or later are required to run Mesa 6.0. It is recommended that you obtain OS version 18 and Executive version 11 as soon as they are available.

The new file creation date standard is now supported. The compiler inserts the creation date of the source file into the .bcd (as does the Binder), and the Debugger checks that the source and object file versions match. The IncludeChecker also makes use of these dates. *Therefore, you must use an editor that supports the file date standard (e.g., Bravo 7.5).*

Language

With two exceptions, Mesa 5.0 source files are compatible with Mesa 6.0. In file names in the DIRECTORY statement, names inside angle brackets are no longer ignored; they are treated as subdirectories of the Alto file system (use of this feature is not recommended). The order of the MACHINE DEPENDENT and MONITORED attributes in record declarations has been reversed; MACHINE DEPENDENT MONITORED RECORD is correct.

Compiler

The order of evaluation in expressions and constructors has been changed, particularly in cases involving imbedded procedure calls; it is no longer left to right. In particular, expressions of the form `word _get[s]*256+get[s]` are suspect, and probably incorrect. For elements smaller than eight bits, the internal representation of packed arrays has changed.

The Compiler no longer supports interactive command input; it reads commands only from the command line, and does not use the keyboard or display (feedback is via the cursor). The command language has been extended and switch processing has changed. The `/c` switch has been deleted; global switches must now be specified with a null file name, e.g. `Compile /p Defs Impl`. Switches are restricted to a single letter. *Do not use complete switch names such as /pause* (each letter will be interpreted as a separate switch). Until the full syntax of the command language is documented, it is best to group all global switches at the beginning of the command line following a single slash.

The log is now written on `Compiler.log`, not `Mesa.typescript`; separate `.errlog` files are still produced. The error log (if any) is deleted if the compilation is successful; conversely, if the compilation fails, the `.bcd` (if any) is deleted.

The implementation of floating point has changed considerably; the IEEE standard format is now used, and the compiler generates calls directly to user microcode (this will produce undefined results if the proper microcode is not present). Calls to software floating point as in Mesa 5.0 can be generated with the `/-f` switch. *Do not use type REAL without first consulting with a floating point wizard.*

Because of bug fixes, previously acceptable programs may no longer compile.

Binder

Like the Compiler, the Binder takes commands only from the command line and does not use the keyboard or display; it writes its log file on `Binder.log`.

Compressed symbols files cannot yet be produced by the Binder.

System

Mesa 6.0 is compatible with XMesa 5.0 microcode version 39 (but some new features are not available with this microcode, e.g. extended memory BITBLT).

Features previously implemented by BasicMesa and XMesa are now a standard part of `Mesa.image`; **MakImage** is no longer a part of the standard system, and must be loaded or bound with the client configuration.

The standard system now supports only command line input; **MesaExec** can be loaded to provide interactive input. The command line switch `/b` can be used to convert the standard system to a basic one. The `/k` switch will disable the allocation of space for the Debugger's bitmap on extended memory machines.

The default maximum numbers of processes (75) and modules (384) have been increased.

Interface changes are described in the System Update memo.

Debugger

The Debugger now requires a strike font named `SysFont.strike` or `MesaFont.strike`; a version of Gacha10 is available on `<AlphaMesa>MesaFont.strike`. A program which produces strike fonts from most `.al` fonts is available (`AlToStrike.bcd`); see also `[Maxc]<AltoFonts>`.

The selection scheme and the assignment of function keys and mouse buttons has changed. Clicking RED once selects a character, clicking twice selects a word, three times a line, etc.; the selection can be extended to the left or right with BLUE. The menu button is now YELLOW (formerly BLUE). FL4 is no longer the stuff key; use FR4 (Spare2) or Keyset2.

Scrollbars no longer occupy a dedicated part of the window, but instead come up on top of the left edge. To obtain a scroll bar, move left just past the edge of the window, then move right slightly, back into the window.

New source window menu commands have been added, and they have been factored into several menus. The Debugger's wisk window has been replaced by a more general Split window menu command. There are now also `Normalize Insertion` and `Normalize Selection` commands.

The interpreter can now be used when in `Display Stack` subcommand mode. Several commands now invoke the interpreter automatically (e.g., `Octal Read: @p, n: SIZE[r]`). The interpreter does procedure calls (the `Interpret Call` command has been deleted).

The constructs LONG, POINTER TO, LOOPHOLE, MIN, MAX, ABS, and open and half open intervals have been added to the interpreter's grammar. Type expressions following % must be enclosed in parentheses. The interpreter syntax **Expression?** has replaced the `Interpret Expression` command.

Break/Tracepoints can no longer be set by typing the source line, and the `Break Module` and `Break Procedure` commands and corresponding `Trace` and `Clear` commands have been deleted; the menu commands must be used. `Clear All Entries/Xits` clears both break and tracepoints. Tracepoints now automatically invoke the normal `Display Stack` command processor (with subcommand `p(arameters)`); the `q(uit)` subcommand (not `b(reak)`) exits to the Debugger's command level, it no longer continues execution of the client. Conditional breakpoints are not yet implemented.

If the source window is loaded with the `s(ource)` subcommand the `Display Stack`, the window will remember the appropriate context for setting breakpoints.

The Debugger no longer ignores case, and the case commands have been deleted; *identifiers must be typed with their correct capitalization.*

Distribution:

AlphaMesa Users
Mesa Group
SDSupport