Enhanced OS-9 Release Notes

Version 3.0

www.radisys.com
World Headquarters
5445 NE Dawson Creek Drive • Hillsboro, OR
97124 USA
Phone: 503-615-1100 • Fax: 503-615-1121
Toll-Free: 800-950-0044

International Headquarters
Gebouw Flevopoort • Televisieweg 1A
NL-1322 AC • Almere, The Netherlands
Phone: 31 36 5365595 • Fax: 31 36 5365620

RadiSys Microware Communications Software Division, Inc.
1500 N.W. 118th Street
Des Moines, Iowa 50325
515-223-8000

Revision C
April 2001

Copyright and publication information

This manual reflects version 3.0 of Enhanced OS-9 for ARM/StrongARM, MIPS, PowerPC, SuperH, and X86.. Reproduction of this document, in part or whole, by any means, electrical, mechanical, magnetic, optical, chemical, manual, or otherwise is prohibited, without written permission from RadiSys Microware Communications Software Division, Inc.

Disclaimer

The information contained herein is believed to be accurate as of the date of publication. However, RadiSys Corporation will not be liable for any damages including indirect or consequential, from use of the OS-9 operating system, Microware-provided software, or reliance on the accuracy of this documentation. The information contained herein is subject to change without notice.

Reproduction notice

The software described in this document is intended to be used on a single computer system. RadiSys Corporation expressly prohibits any reproduction of the software on tape, disk, or any other medium except for backup purposes. Distribution of this software, in part or whole, to any other party or on any other system may constitute copyright infringements and misappropriation of trade secrets and confidential processes which are the property of RadiSys Corporation and/or other parties. Unauthorized distribution of software may cause damages far in excess of the value of the copies involved.

April 2001

Table of Contents

Chapter 4: PowerPC Notes 31

Chapter 5: SuperH Notes 47

Chapter 1: Introduction

As part of Microware's policy of continued product development, this Enhanced OS-9 Version 3.0 represents a maintenance and update release to incorporate all the improvements that have been introduced into the component parts. Two of the more significant upgrades include the following:

- **Threads support.** OS-9 version 3.0 includes support for POSIX-style threads. In the OS-9 implementation, POSIX threads are lightweight processes. Each thread behaves like a process but has a much lower overhead in terms of system resources. The kernel uses one resource descriptor for each process and one state descriptor for each thread. The state descriptors contain only the information necessary to maintain and schedule a thread of execution. For more information about the Microware implementation of POSIX threads, see the ***Using OS-9 Threads*** manual.

Note

If you are upgrading from a previous version of OS-9, please be sure to read the **Migrating existing OS-9 code to OS-9 Version 3.0** section in Chapter 7: Processor Independent Notes.

- **Support for Java 3.1.** This new release of PersonalJava Solution for OS-9 includes support for native threads and the Java 2 security model. It can also run MAUI applications and allows MAUI applications to run applets. For more information about the Microware support for Java, see the ***Using PersonalJava Solution for OS-9*** and the ***Using JavaCodeCompact for OS-9*** manuals.

These release notes cover the changes made to the Enhanced OS-9 package in the time since the previous release. If upgrading from an earlier version, these release notes should be read in conjunction with the relevant older release notes.

Note

Files for Enhanced OS-9 Version 3.0 are updates to previous product releases. Although these files are intended to be installed on top or your existing version of OS-9, it is recommended that you complete a backup of your system before installation.

Chapter 2: ARM/StrongARM Notes

This chapter provides an overview of the changes and improvements made to Enhanced OS-9 for ARM/StrongARM.

Enhanced OS-9 for ARM/StrongARM Version 3.0

ARM/StrongARM Resolved Problems

General Resolved Problems

- CF7351: Low-level ethernet driver was not functioning on the SideArm board when using `undpd`. The problem was that the `lle509` driver was not properly initializing the interrupt register on the SA1101 companion chip. This problem has been fixed with edition 16 of `lle509`.
- CF8052 - The `llne2000` low-level ethernet driver did not work on the Assabet using `undpd`. This problem has been fixed with edition 5 of `llne2000`.
- CF8062: The LCD driver on the Assabet does not clear the screen after being deinized. Now, the `gx_sa1100` graphics driver will clear the screen to black before shutting down.
- CF8165: When building a bootfile with the wizard for any StrongARM board, the OS-9 utility `alias` may be included more than once. This has been fixed.
- CF8227: Every time a system-state application exits, 20 bytes of memory would not be reclaimed by the system. This issue has been fixed by edition 25 of `SSM`.
- CF8236: On a Sidearm/Sidekick board, occasionally mouse interrupts would be lost. Intel confirmed that there is a window on the SA1101 companion chip where interrupts would be lost. This issue will be fixed in the SA1111 chip. A workaround exists in software by using edition 5 of the `irq1101` module.

Compiler Resolved Problems

These resolved problems are for the Microware Ultra C/C++ compiler version 2.4.

- CF8193 - The ARM assembly optimizer no longer has the potential of putting PC-relative data earlier in the code than the reference. Although it does not cause a problem, it is indicative of a problem with the optimizer that could cause compile failures.
- CF8911: The ARM code generator no longer fails to generate correct code for particular structure casting expressions.

ARM/StrongARM Known Issues

- On SideARM boards, RomBug will not fit into the coreboot without exceeding the board's flash size limitation.
- `msginfo -d` reports error 10:45. The default MAUI configuration does not support `msg_set_filter()`. Systems may be configured with `mauidrvr_filter` driver instead of `mauidrvr` to support `msg_set_filter()`. The `mauidrvr_filter` driver is a bit larger and slower than `mauidrvr`, therefore is not the default. The following is an example of an error that can occur using the `-d` option with the default MAUI configuration.

```
Super:msginfo mp_mbox -d
Mailbox status for 'mp_mbox'
  number of entries : 25
  free entries : 25
  message size : 132
  link count: 3
MAUI Fatal(22|22): 10:45 detected in msg_set_filter.
Error #10:45 `Failed to install filter function for dump.
```

- The jview MAUI demo on a StrongARM aborts with an error reading JPEG images wrapped in a data module. Normal JPEG files work.

- On some StrongARM hardware, the USB Device software may respond better to the HawkView program when running the StrongARM core at 191MHz, instead of the 206 (which is the default value).

  To set the processor speed to 191MHz, edit the `systype.h` file in the port directory (i.e. `MWOS\OS9000\ARMV4\PORTS\ASSABET\systype.h`). Search for `S_191_32`, and comment out the define for `S_206_32` and un-comment the `S_191_32` define. Below is an example.

```
/******************************
 * Processor Speed + Dram Size *
 *********************************/
/* #define S_133_32 */ /*132.7Mhz PLL and memory timings +(32Mb SDram)*/
#define S_191_32   /* 191.7Mhz PLL and memory timings +(32Mb SDram) */
/* #define S_206_32   /* 206.4Mhz PLL and memory timings +(32Mb SDram)*/
```

  After modifying the `systype.h` file, rebuild the port by typing the following:

```
$ cd OS9000/ARMV4/PORTS/ASSABET
$ os9make clean purge all
```

  This is only possible if via an OEM OS-9 package. The processor speed is not adjustable via a BLS package.

Enhanced OS-9 for ARM/StrongARM Version 2.0

Threads Support Available

The latest version of the OS-9 kernel (version 125 or newer) supports multiple threads.

For More Information

See **OS-9 Operating System Notes** on page 151.

Enhanced OS-9 for ARM/StrongARM Version 1.2

General Improvements

The following general improvements were made to Enhanced OS-9 for ARM/StrongARM:

- The Microware IDE (Hawk) has been updated to use a new, faster protocol to communicate with the target processor. Quicker code tracing and animation sessions lead to faster application debugging sessions.
- Hawk support for simultaneous debugging of multiple target applications is easier to use. It is also possible to simultaneously debug two or more applications on different target hardware (even if they are different processors). This facility is particularly useful when the target systems communicate over a network.
- The TCP/IP and UDP/IP stacks are considerably faster.
- Generic and specific compiler optimization improvements have benefited all processors.
- Java is updated to V3.0.1 (Java is an optional add-on package).
- Software floating point emulation has been completely rewritten to make it faster.
- The following new board level solutions are added to the ARM package:

  ADS Graphics Client

  Intel Assabet SA1110

  PID7T (ARM7T)

- Source code for the following new board level solutions are added to the ARM/StrongARM package:

  ADS Graphics Client

  Intel Assabet SA1110

  PID7T (ARM7T)

  Penpal I

- The following new add-on packages are available:

  X.25/LAP-B WAN support

  TrueFFS flash support

  ISDN (Q.931)/LAP-D support

  SNMP

Note

The StrongARM Brutus board is no longer available. The Brutus code in the OEM package and the Brutus Board Level Solution will be relegated to unsupported status in the next package release.

ARM/StrongARM Enhancements

- Newly supported PCMCIA cards for ARM/StrongARM targets. The ARM/StrongARM PCMCIA's code now supports and has been tested with the following NEW cards. This is in addition to previously supported cards.

  SOCKET-LPE ethernet card (NE2000 compatible)

  3COM Megahertz Model 3CCE589ET

  Megahertz 14400 Fax modem cards

  Motorola Montana 33.6 Fax Modem

Intel Value Series 200 Linear Flash (with `pflash` utility and `fcopy` booter)

- Support added for ThinClient/GraphicsClient on-board flash. The `pflash` utility was added to the ThinClient and GraphicsClient ports. This utility allows for the burning of an OS-9 ROM image into the on-board Intel StrataFlash parts. The utility also supports the burning of Intel Value Series 200 linear flash cards, which can be booted from using the `fcopy` booter.

ARM/StrongARM Resolved Problems

Compiler Resolved Problems

- CF3078: `optarm`—remove gratuitous labels to allow more optimizations. The assembly optimizer now deletes the destination label used by a branch around an instruction. For example:

```
...
ldr r7,DATA
b AROUND ***skip
DATA dc.l x
AROUND
...
```

This enables basic blocks to be larger, essentially allowing the code before and after this chunk of code to be part of the same basic block. This has been fixed in Edition 62 or later of `optarm`.

- CF5693: `iopt` supports these processor-specific optimizations. ARM CSEfys global addresses, most floating point constants, and large immediates (not representable with two instructions). PowerPC, SPARC, and MIPS CSEfy floating point constants (as they must be loaded from memory), and large immediates not representable in one instruction. Edition 93 of `iopt` supports these optimizations.

- CF5704: beARM bad codegen (incorrect reuse of register). Use of the diveq (/=) operator may result in incorrect reuse of a temporary register. This has been fixed in Edition 59 or later of bearm.
- CF5744: bearm does not function correctly with some constants. The ARM architecture allows 8-bit immediates for general-purpose arithmetic. In addition, these 8-bit immediates can be rotated by 2*n bits. This allows you to specify 0xff as well as 0xff00 and 0xf000000f.

  In general, the back end is good at using immediates to specify constant values. However, it is not good with the pathological cases; specifically those where a rotation, not just a shift, is necessary to specify the immediate. For example:

  ```
  0xfc000003
  0xf000000f
  0xc000003f
  ```

  Edition 59 or later of bearm fixes this problem.
- CF6292: RFI compiler should generate statically linked FPU emulation code. The emulation of floating point instructions on targets that do not support floating point operations in hardware can be very expensive. In addition to the time to emulate the operation, there is the added overhead of the exception and the decoding of the instructions. The compiler should support software emulation.

  With version 2.3, Generic Floating Point Software Emulation is supported. To specify "software emulation" of floating point instead of the standard hardware emulation, use the ,fp target suboptions (for example -tp=ppc,fp). Edition 67 or later of xcc, Edition 2 or later of ficode, and appropriate libraries are required.
- CF6401: bearm does not respect some volatile expressions. The ARM compiler is not respecting some seemingly gratuitous loads involving volatile expressions.

  In addition, bearm has been updated to emit an error (or a warning, when using the -bepv compiler switch) if a short volatile is used when targeting an ARMv3 target. Edition 60 or later of bearm fixes this problem.

- CF6442: beARM should accept __reg_lr() and not __reg_ra().

  Given the following code:

  ```
  _asm(" ", __reg_lr());
  ```

  The ARM back end will error out with a pattern match error. However, with the following code:

  ```
  _asm(" ", __reg_ra());
  ```

  Everything, seemingly, will work. __reg_ra() is the incorrect alternative name for accessing R14. It should be __reg_lr(). Edition 60 or later bearm fixes this problem.

- CF6551: ARM compiler may generate incorrect code for certain signed/unsigned compares. Edition 61 or later of bearm fixes this problem.

- CF6563: optarm optimizes some compares with zero incorrectly. optarm will optimize the following code:

  ```
  add r7,r7,r8
  cmp r7,0
  bhi label
  ```

  Into the following:

  ```
  adds r7,r7,r8
  bhi label
  ```

  This is wrong. The problem is that the cmp and adds instructions set the CARRY flag slightly differently. ADDS sets the carry flag in the same manner as CMN R7,0 would. CMP R7,0 and CMN R7,0 are not equivalent.

  The assembly optimizer should not perform this optimization if the conditional involved uses the CARRY flag at all. I.e., CS/HS, CC/LO, HI, and LS are off limits. This only pertains to ADD and ADC. The SUB instructions generate CARRY the same as CMP. Edition 63 or later of optarm fixes this problem.

- CF6694: aarm incorrectly generates 90 immediates. The ARM assembler incorrectly generates certain immediates. This only affects certain immediates > 0x10000000. Edition 55 or later of aarm fixes this problem.

- CF7032: `optarm` doesn't recognize `ldm/stm` mask expression using `|`. The back end may emit a `load/store` immediate that looks like the following when doing structure copies:

  ```
  ldmia r7!,0x100|0x200
  stmia r7!,0x100|0x200
  ```

  However, the assembly optimizer is not set up to handle this. Edition 65 or later of `optarm` fixes this problem.

- CF7057: `optarm` may cause linker error. On very rare occasions, it is possible for pc-relative data to be moved too far from it's usage, resulting in a linker error complaining about a value being too large for a field. The workaround is to use the `-aop=p` switch. Edition 65 or later of `optarm` fixes this problem.

- CF7223: Improved floating point software emulation for ARM. The ARM versions of the `fp` software emulation libraries (`fpl.l`, see 6292) have been optimized. In addition, the ARM `fp` hardware emulation module (`fpu`) has been optimized and improved (better accuracy and support for "special" values).

- CF7593: `bearm` bug-reg used for function pointer conflicts with link register. The problem is that the register used to hold a function pointer turns out to be R14. Therefore, when the link register is set up as part of the call through the pointer, the function pointer is trashed:

  ```
  *      cvfcn = awtImage->convert[flags];
  add R7,R6,(=awtImage)&0xfffff000
  ldr R7,[R7,(=awtImage)&0xfff]
  add R8,R7,48
  ldr R14,[R8,lsl(R7,2)]
  *      ret = cvfcn(cmh, x, y, w, h,
  mov R7,R5
  mov R8,R0
  mov R9,R1
  mov R10,R3
  ...
  add R11,R6,(=awtImage)&0xfffff000
  ldr R11,[R11,(=awtImage)&0xfff]
  add R11,R11,16
  str R11,[R13,44]
  mov R14,R15
  mov R15,R14
  ```

The function call is extremely fast. It simply goes to the next instruction. Edition 62 of bearm fixes this bug.

OS-9 Resolved Problems

- Issue: Kernel debug assist code could corrupt r0. A bug was fixed in the ARM specific kernel where the register value of r0 could be corrupted while actively debugging a process.
- Issue: SSM—Vectors now can be virtually remapped from a ROM which is at physical address 0, into DRAM at virtual address 0. This speeds up interrupt response time by allowing interrupts to be dispatched from fast burstable DRAM, instead of from a slow non-burst flash. This re-mapping is triggered by defining the first cache list entry as a sub 1 Meg size. This remapping is also useful in working around LCD bandwidth problems.
- Issue: SSM—The CM_WB mode was added to the definable cache types. This allows for a regions to be defined as write buffer only.
- Issue: SSM—The ARM SSM's transadd() function now returns SUCCESS if virtual translation is disabled.

  (Only the Brutus board port actively uses virtual memory translation by default).

ROM Component Resolved Problems

- CF2570: dbgserv—Debugger incorrectly showing data aborts as originating at address 0. The ARM ROM code now will set bit 0 of the cc_hwflags to flag if CP15 is accessible. This provides the debugger with a safe way to know it can access coprocessor 15 registers on ARMs which implement it. The CP15 values are now viewable from both ROMBUG and HAWK.
- Issue: Dbgentry—Fixed incorrect CPSR setting. When calling the debugger prior to the kernel, an incorrect CPSR value was being set. This caused a boot to intermittently fail on StrongARM and would put ARM7T cores into thumb mode.

Board-Specific Resolved Problems

- CF2779: Brutus PCMCIA ethernet drivers did not work using the twisted pair connection. The current high- and low-level Ethernet drivers now work with either twisted pair or BNC connections. An incorrect part initialization caused the failure of twisted pair.
- CF5709: Corrected "references" spelled wrong in `registry.rul` script for StrongARM installs.
- CF6611: UCB1200 Driver fails with `light` touches on Brutus. The `ucb1200` touch screen driver changed its internal sequencing of touch plate biasing to close a hole where the hardware would not allow clearing of the interrupt, resulting in a system crash.
- Issue: The `UCB1200` driver now allows for over sampling. The `ucb1200` driver now gets an over sampling factor from its descriptor. This averaging will help in the removal of bad sample points resulting in a smoother interface.
- Issue: The `UCB1200` driver now makes use of the `UCB1300` pressure mode suggestions. The `ucb1200` driver now switches to pressure mode prior to changing to position mode. This change reduces bad data points generated by the internal ADC during its settling. This change is as documented in the `ucb1300` manual.
- CF7044: Boot process will hang if ATA PCMCIA card is not present. All ARM board ports now implement an ATA time out which allows the boot process to bypass non-occupied sockets.
- Issue: The `SA11x0 pic` library enhancements: The `SA11x0 pic` library now sets interrupt triggering based on a `systype.h` settings if defined. Also, the library now correctly sets triggering on the shared GPIO lines, if used as interrupts, and will set up direction and functions if a GPIO is used as an interrupt.
- Issue: The StrongARM BSP specific PCMCIA parsing code fix. The StrongARM BSP specific PCMCIA parsing code did not correctly handle the `NOLINK` tuple. The fix allows a wider variety of PCMCIA cards.

ARM/StrongARM Known Issues

- CF8062: The SA1100 LCD driver `gx_sa1100` should clear the display when an application closes its path to the graphics device.
- CF8052: User-state debugging (`undpd`) over the low-level networking modules is non-functional on the Assabet with the `llne2000` driver. Workarounds include the following:

  Use high-level networking (i.e., SPF) and associated daemons (`spfndpd`) for user-state debugging.

  When doing system-state debugging, modules should already be present on the target or can be downloaded using kermit.
- Booting with bootp is not supported on the ADS Graphics Client board. Some versions of the graphicsclient may have trouble transmitting data using the SP91C94 driver. At release time it appears to be a problem with one particular board but this has not been confirmed. The one board that is failing does have the oldest version of the SMSC LAN91C96 chip and it is possible other early chips have the same problem. The symptom of the problem is the board will hang when quickly transmitting large amounts of data.
- The msginfo binary is not included with the other maui demo programs in /mwos/OS9000/ARMV4/CMDS/MAUIDEMO. The source is included and this module can be built from /mwos/SRC/MAUI/DEMOS/MSGINFO.
- The aurecord program will not detect a default sample rate or default number of channels and they must be specified on the command line.

  For example, the following will work:

  ```
  aurecord -e1,7966 -m -t10 test2.au
  ```

Chapter 3: MIPS Notes

This chapter provides an overview of the changes and improvements made to Enhanced OS-9 for MIPS.

Enhanced OS-9 for MIPS Version 3.0

MIPS64 Resolved Problems

Compiler Resolved Problems

These resolved problems are for the Ultra C/C++ 2.4 compiler.

CF8115: `xcc` and `bemips`, the MIPS back end, were changed to reflect new MIPS nomenclature. Now the option `-tp=mips64` is used instead of `-tp=mips4000` with `xcc`, and `-pmips64` is used instead of `-pmips4000` with `bemips`. Specific 64-bit MIPS processors (e.g. IDT4650), remain unchanged.

Threads Support Available

The latest version of the OS-9 kernel (version 125 or newer) supports multiple threads.

For More Information

See **OS-9 Operating System Notes** on page 151.

Enhanced OS-9 for MIPS Version 1.0

MIPS Port Information

This release of Enhanced OS-9 for MIPS contains two Wizard configurations. The 79S465_4650 configuration covers the IDT4650/4640 processor on the 79S440 daughter board. The 79S465_4700 covers all other configurations including the native IDT4700 processor and the IDT64575/64574 processors on the 79S574 daughter board.

For the 79S465 reference platform, the Wizard has two memory configurations. The **default** memory setting is used to boot the ROM file from the IDT SIM. The **8 Meg Remote** setting is set up in order for the IDT SIM to boot coreboot, then coreboot can be used to boot the bootfile via bootp or SCSI.

Chapter 4: PowerPC Notes

This chapter provides an overview of the changes and improvements made to Enhanced OS-9 for PowerPC.

Enhanced OS-9 for PowerPC Version 3.0

General Improvements

The following general improvements were made to Enhanced OS-9 for PPC:

- A new OEM product for MP5 has been added.
- A new board level solution for the IBM 450GP has been added.
- The USB Peripheral SDK has been added.

PowerPC Resolved Problems

General Resolved Problems

- CF2532: `F_CHKMEM` returns `E_BusErr` instead of `E_BPADDR`. This was a bug in the PPC8xx SSM. Fixed in edition #30 of `ssm801`, `ssm821`, and `ssm823`.
- CF2534: `F_CHKMEM` returns `SUCCESS` for invalid range coverages. This problem was a bug in the PPC8xx SSM. Fixed in edition #30 of `ssm801`, `ssm821`, and `ssm823`.
- CF8100: OS-9 does not start on FADS860T (70 vs. 90 ns FLASH issue). Previous version didn’t support 70ns Flash. The latest version supports this part and other parts that are not supported in the original FADS port. The flash size default has also been changed to support a wider variety of parts.
- CF8260: DEC21140 Driver fails MULTICAST test. Multicast support is added in Edition 28 of the driver. The basic functionality has also been fixed.
- CF8303: Corrected `isr.c` in `spquicc` driver. The status checking in `xmit_irq` has been fixed. Edition 21 of this driver reflects the change.

- CF8960: The pccard module or 860FADS is not present. The module is not part of the port, so the PC Card IDE option was disabled in the Wizard by changing the ROM_PCMCIA_IDE macro to FALSE in the 860tfads.ini file.
- CF9498: Change to /mwos/src/rom/serial/cpm/iocpm.c. The _pic_enable and _pic_disable functions have been changed to _PIC_ENABLE and _PIC_DISABLE, so they can be independently used across different platforms. The baud rate table now supports up to 115200 bps. The changes are reflected in Edition 24 of the driver.
- CF9966: Fixes for SCCPM (MPC8xx). For the 8260 using the SCC serial port, the driver will initialize the right registers. Edition 24 of this driver reflects the change.
- CF9967: Modification in the SPQUICC code (MPC8xx). The driver now only invalidates the cache for the received length instead of the complete buffer. It will also allocate new mbufs if there is not enough space in front of the packet. Edition 25 of the driver reflects these changes.
- CF10269: SCSI disks hang during high data load, disconnect during data transfer. The controller's scripts (v53c810.ss and v53c710.ss) have been updated to eliminate this problem.
- CF10317: MBX8xx\ROM\default.des bug (comm_c vs. c_comm). The MBX8XX\ROM\default.des file now includes the right macros (c_cons and c_comm instead of cons_c and comm_c).

Compiler Resolved Problems

These resolved problems are for Ultra C/C++ 2.4.

- CF8519: The optimization of "for" loops on the PowerPC has been corrected such that optimized code is emitted for all types of loops.
- CF7775: optppc - macro definition should at least be respected. The problem was that the scheduler was mistakenly scheduling Directives by including them in basic blocks. Basic blocks should end before a Directive. This problem is fixed in Edition 67 of optppc.

Enhanced OS-9 for PowerPC Version 2.0

Threads Support Available

The latest version of the OS-9 kernel (version 125 or newer) supports multiple threads.

For More Information

See **OS-9 Operating System Notes** on page 151.

Enhanced OS-9 for PowerPC Version 1.4

General Improvements

The following general improvements were made to Enhanced OS-9 for PowerPC:

- The Microware IDE (Hawk) has been updated to use a new, faster protocol to communicate with the target processor. Quicker code tracing and animation sessions lead to faster application debugging sessions.
- Hawk more easily supports simultaneous debugging of multiple target applications. It is also possible to simultaneously debug two or more applications on different target hardware (even if they are different processors). This facility is particularly useful when the target systems communicate over a network.
- The TCP/IP and UDP/IP stacks are considerably faster.
- Generic compiler optimization improvements have benefited all processors.
- Java is updated to V3.0.1 (Java is an optional add-on package).
- Software floating point emulation has been completely rewritten to make it faster.
- All software has been recompiled with an improved optimizing compiler, leading to faster tighter code.
- Software floating point emulation has been completely rewritten and is now 2-5 times faster (depending on exact application).
- A new compiler option allows floating point to be compiled into the application, removing the need to take an exception into system state (where speed is crucial, and memory space is available).
- This package supports the Motorola 8240, Motorola 8260 and IBM 403 for the first time.

- The following new board level solutions are added to the PowerPC BLS package:

  Motorola MTX603/4

  EST8260

  Motorola 8240 SandPoint

  Embedded Planet RPXLite 850

- Source code for the following example board level solutions are added to the PowerPC OEM package:

  Motorola MTX603/4

  EST8260

  Motorola 8240 SandPoint

  Embedded Planet RPXLite 850

  IBM 403

  MVME1603/4

- The following new optional add-on packages are available:

  X.25/LAP-B WAN support

  TrueFFS flash support

  ISDN (Q.931)/LAP-D support

Note

Motorola does not recommend the PowerPC 821 for new designs. Therefore, the PowerPC 821 code in the OEM package and the MBX821 Board Level Solution will be relegated to unsupported status in the next package release.

PowerPC Resolved Problems

- CF6108: Removed incorrect and unused defines in the 8XXXFADS `sysinit.c`.
- CF2456: Previously, the PowerPC 403 clock tick driver (`tk403ga`) could miss an interrupt.
- CF7207: Previous versions of OS-9 for PowerPC did not allow user applications to trap an FPU exception such as divide by zero. The latest version does allow these exceptions to be trapped and passes the PowerPC Floating Point Status and Control register (`fpscr`) value to the trap handler to indicate the exact exception condition. The `fpscr` value is passed as a `var args` argument. An example handler for this exception is as follows:

```
u_int32 zerdiv_except_func(u_int32 v, u_int32 pc, u_int32 a, ...)
{
   va_list list;
   va_start(list, a);
   printf("Vector = %d, PC = %X\n", v, pc);
   printf("\tfpscr value = 0x%08x\n", va_arg(list, u_int32));
   ...
}
```

- CF2812: Non-destructive stack checking requested for PowerPC. For PowerPC, use the `-bepnd` compiler switch to enable non-destructive stack checking. This has been implemented in `beppc`, Edition 59 or later. It requires the latest `clib/csl` library (which contains the `stkhandler_nd` routine).
- CF2956: `sp21140` - wrong device id if bus and device specified. Edition 22 or later of `sp21140` fixes this problem.
- CF4370: `beppc` - some callee-saved registers are not saved correctly. Some callee-saved registers were not being saved correctly, which could result in unexpected results. Edition 55 or later of `beppc` fixes this problem.
- CF4755 issue: `optppc` - invalid copy propagation optimization. Given the following code:

```
nop
ori r0,r3,0
stbx r0,r0,r5       * stbx r3,r0,r5
andi. r0,r0,65535
nop
```

optppc (ed 46) will incorrectly change the stbx instruction (while doing copy propagation):

```
* UCCaoPPC
 nop
 stbx r3,r3,r5 * stbx r3,r0,r5
 andi. r0,r3,65535
 nop
```

The second operand in the stbx instruction must remain r0, because r0, in this case does not mean r0, it means constant zero. This has been fixed in Edition 48 optppc, and improved in Edition 51.

- CF5425: some spr's are not recognized correctly by appc. The PowerPC assembler does not recognize the following spr's correctly:

```
 fpecr : 1022 (505)
 pmc4: 989 (750)
```

pmc4 is missing as a result of a typo -- it is available as pcm4 (the c and m are swapped). fpecr is not there. These should be added to the grammar.ppc file and the assembler remade.

Also, the following are decoded incorrectly:

rtcu and rtcl

The 601 has these defined two different ways -- depending upon whether writing is done or reading. Four and five are the numbers for User-level reading and 20 and 21 are used for supervisor-level writing.

The following SPR's were made into Read-Only Registers:

Table 4-1 Read-Only Registers

NAME	NUM	APPLICABLE PROCESSORS
rtcu	4	601 (user)
rtcl	5	601 (user)
dc_dat	570	821
dear	981	403

Table 4-1 Read-Only Registers

NAME	NUM	APPLICABLE PROCESSORS
`ecr`	148	505
`icr`	148	821
`icdat`	562	505
`ic_dat`	562	821
`pvr`	287	403,505,601,821
`tbhu`	972	403GC
`tblu`	973	403GC

The following SPR's were made into Write-Only registers:

Table 4-2 Write-Only Registers

`rtcu`	20	601
`rtcl`	21	601
`eie`	80	505,821
`eid`	81	505,821
`nre`	82	505
`nri`	82	821
`tbl`	284	505,603,821
`tbu`	285	505,603,821

The following DCR register was made into a Read-Only register:

Table 4-3 DCR Read-Only Register

bear	144	403

If any of these registers are used inappropriately, an error is not guaranteed. Instead the name will be treated as an undefined symbol. Only if no label (e.g., code or data location, equ or set definition) is found with that same name will an error be generated by the linker.

The following are now correctly recognized:

Table 4-4 Correctly Recognized Registers

`fpecr`	1022	505
`pmc4`	989	750

Edition 56 or later of the PowerPC assembler (`appc`) has the fix.

- CF5693: `iopt` support for processor-specific optimizations. ARM `CSEfys` global addresses, most floating point constants, and large immediates (not representable with two instructions). PowerPC, SPARC, and MIPS CSEfy floating point constants (as they must be loaded from memory), and large immediates not representable in one instruction. Edition 93 `iopt` supports the CSEfying optimization for ARM, PowerPC, SPARC, and MIPS.
- CF5832: `optppc` - support instruction/list scheduling. Target-driven instruction scheduling has been added to the PowerPC assembly optimizer.

  The "scheduling methods" switches were updated (adding `-sw` and `-st`):

  `-s<method>` set the peephole scheduling method, where `<method>` is:

  `s` = spread dependent instructions

  `c` = compress floating point instructions

  `t` = target-driven scheduling

  `w` = target-driven scheduling w/ compression

  `n` = no instruction scheduling

  Edition 60 or later of `optppc` has this enhancement.
- CF5912, CF6225: RFI - `beppc` and `optppc` should support register freelists/renaming. Edition 58 `beppc` (PowerPC backend) generates register free lists.

These can be used by the Edition 60 optppc to improve optimizations:

- improved copy propagation, since better register liveness info is known
- improved scheduling possibilities with the addition of register renaming.

- CF6029: `beppc` - does not generate `mulli` instructions.

  Given the following code:

  ```
  int mul_13(int x)
  {
      return x * 13;
  }
  ```

  The PowerPC backend will generate:

  ```
  *       return x * 13;
   addi r0,r0,13
   mullw r3,r3,r0
  ```

  This is one instruction more than necessary. It should be able to do:

  ```
   mulli r3,r3,13
  ```

  Edition 56 `beppc` will generate `mulli` instructions when appropriate.

- CF6031: Support 21143 Ethernet Hardware (e.g., on the MTX070). Support for the 21143 device has been added and will be available with edition 22 of the `sp21040/sp21140/sp21143` driver. The descriptor edition was bumped to 19.

- CF6184: Bad register usage in PowerPC `clib.il:__asm_fmove/memcpy.`

  It is possible that the i-code in the `clib.il` library for `__asm_fmove` will generate bad code. Specifically, it might use "r0" as follows:

  ```
          addi r0,r0,-4
  ```

  When the intended action is to decrement r0 by 4, not move -4 into r0.

This is a result of incorrect use of the `__reg_gen() _asm` pseudo-function in the `__asm_fmove` code. Instead, `__reg_base()` should be used! The latest release of the `clib` library fixes this problem.

- CF6220: compiler could generate faster divide-by-const by mult-by-reciproc. The generated compiler code now generates code that performed division-by-constant as a multiply by reciprocal. Edition 57 or later of the PowerPC backend supports this feature.
- CF6234: `optppc` - incorrectly removes register copy. During the course of copy propagation, `optppc` will incorrectly delete the initial copy in the following example:

```
nop
mr r4,r5
rlwimi r4,r6,1,2,3
mr r5,r4
nop
```

  Generating:

```
nop
rlwimi r4,r6,1,2,3
mr r5,r4
nop
```

  The problem is that `rlwimi` is an `INSERT` instruction, it and `rldimi` are the only two in the PowerPC instruction set. The destination register (in this case r4) is both a destination and a source register. An appropriate translation of the original code would have been:

```
nop
rlwimi r5,r6,1,2,3
nop
```

  Edition 56 or later of `optppc` fixes this problem.

- CF6292: RFI - compiler should generate statically linked FPU emulation code. As the emulation of floating point instructions on targets that do not support floating point operations in hardware can be very expensive (in addition to the time to emulate the operation, there is the added overhead of the exception and the decoding of the instructions), the compiler should support software emulation.

With version 2.3, Generic Floating Point Software Emulation is supported. To specify "software emulation" of floating point instead of the standard hardware emulation, use the ",fp" target suboptions (e.g, -tp=ppc,fp). Edition 67 or later of `xcc`, Edition 2 or later of `ficode`, and appropriate libraries are required.

- CF6293: Improve PowerPC FPU implementation. The PowerPC Floating Point hardware emulator has been optimized to provide improved performance. It is also smaller. Edition 23 or later of `fpu403`, `fpu602`, `fpu821` FPU modules are affected.
- CF6334: FPU emulation: gives differing results than hardware does. The FPU module does not perform round-to-nearest (even) properly for single precision multiply (at least). Given the following numbers:

  0x409c71c7 (4.888888836)

  *0x41400000 (12.0)

  The result should be: 0x426aaaaa, but it is 0x426aaaab.

  The fractional parts are:

  0x9c71c7

  * 0xc00000

  Resulting in: 0x7555 0x554400000

  Normalized (with the second word being the rounding word) this would be:

  0xeaaaaa 0x80000000

  This should only round if it would round to even. The result is already even so no rounding should occur. Edition 23 or later of `fpu403`, `fpu602`, `fpu821`, etc. fixes the problem.
- CF6383: `optppc` - in some cases treats `sp` and `r1` as different registers. The PowerPC `asmopt` will, in some cases, not recognize that `sp` and `r1` are the same register. Examples are loc tracking and copy propagation (where an internal error can be generated because `strcmp("sp","r1") != 0`. Could also cause problems with some copy propagations and register renames (e.g., if we are replacing `r1` with something and don't recognize `sp` as `r1`....). Edition 60 or later of optppc fixes this problem.

- CF6963: MBX8xx wizard - selecting `fdraw`, `fcopy` is used instead. The `fcopy` and `fdraw` radio buttons now select the correct files for inclusion in a boot.
- CF7184: 8XXFADS - `sysinit.c`: writing to ICR doesn't do anything. In the `sysinit.c` code for the 8XXFADS port (located in `8XXFADS/ROM/ROMCORE`), there was the following:

  ```
   xor r0,r0,r0 clear register
  ...
   ifndef EXTDBG
    mtspr der,r0
    mtspr icr,r0
   endc
  ```

  The problem is that the write to ICR doesn't do anything at all (much less clear ICR as is intended). From the MPC860 PowerQUICC manual:

  37.5.2.1 Interrupt Cause Register (ICR)

  The ICR indicates the reason that debug mode was entered. ICR bits are set by the hardware and cleared when the register is read. Attempts to write to ICR are ignored.   All bits are cleared when exiting reset.

  Instead of a write, a read should be performed. This also affects any board ports BASED on the 8XXFADS port.
- CF7797: Ethernet is not working on a PPC860T Rev.D. Edition 4 `sp860t` fixes this problem.(The descriptor edition changed to 4 as well). Edition 3 `ll860t` fixes this problem.
- CF7420 - The `SPQUICC` driver now limits the number of transmit packets that are queued waiting to be sent. The queue size is descriptor configurable and defaults to 50. The cache flush for packets being transmitted has been moved out of the interrupt service routine. This reduces the amount of time spent in interrupt context.
- CF2535 -- Fixed `F_CHECKMEM` zero sized boundary case in PPC8XX SSM. The PPC8XX SSMs were fixed to return success for a 0 byte sized checks.

- CF6311 -- Baud rate of 600 was not functional using the SCCPM driver with high clock speed. The calculation for the baud divider calculation for 600 baud would overflow with system clocks speeds above 40Mhz. The problem was corrected.
- CF7756 -- Slowness of new REV D 8xx PowerPC parts. The PPC 8XX SSM module was updated to correctly recognize the newest 860p and 860T parts. Prior to edition #29, the SSM module did not identify these parts and would use a compatibility setting for early silicon. This resulted in a 50% loss of performance. Edition #29 or greater fixes the problem.

OS-9 Resolved Problems

- FastBoot technology was introduced into the PPC kernels.

Power PC Known Problems

- On the Sandpoint 8240 system, the serial consoles may lose characters if the hard disk is active at the same time characters are being received.
- On the FADS boards, low-level PCMCIA booting is not supported. Selecting these options for your FADS board in the Configuration Wizard will cause the build to fail.
- CF8102—`hlproto` causes the SBC8260 to crash. The Protoman interface trap module (`hlproto`) for user-state connections over low-level drivers has been removed from the default P2 list. This module causes the SBC8260 to crash.

  This module is included in Configuration Wizard-created boots that use Ethernet low-level debugging or include non-SPF user-state debugging in the bootfile.

Chapter 5: SuperH Notes

This chapter provides an overview of the changes and improvements made to Enhanced OS-9 for SuperH. It covers both the SH-3 and SH-4 processor families. The following sections are included:

- **Enhanced OS-9 for SuperH Version 3.1**
- **Enhanced OS-9 for SuperH Version 3.0**
- **Enhanced OS-9 for SH-4 Version 2.2.9**
- **Enhanced OS-9 for SH-3 Version 2.2**

Enhanced OS-9 for SuperH Version 3.1

Configuration Wizard

- CF10454: The Wizard cannot build `pflashrom` from rom with SH-4 v1.1, but can in SH-3 1.1.

  This problem is caused by improper CODO and DEL of ROM in the makefile in the following location: `MWOS/OS9000/SH4/PORTS/SH7750SE/BOOTS/INSTALL/PORTBOOT`.

Enhanced OS-9 for SuperH Version 3.0

General Improvements

- Support for threads have been added.
- General bug fixes and improvements have been made.

SuperH Resolved Problems

SH-3 General Resolved Problems

- CF10197: Power management causes the system to hang for SH3.

  The problem is with the `sc7709` serial driver and a descriptor/ driver header mismatch. The descriptor did not have the proper power management fields made into it, while the driver did. Mismatch has been corrected, and all serial drivers have been checked and fixed where appropriate for this possible error.
- CF10282: Incorrect bus controller setting for SH7709A solution engine board.

  The `sysinit` code for the SH7709A solution engine board incorrectly put the BCR2 initialized data in to BCR3 register. This has been fixed.

Threads Support Available

The latest version of the OS-9 kernel (version 125 or newer) supports multiple threads.

For More Information

See **OS-9 Operating System Notes** on page 151.

Enhanced OS-9 for SH-4 Version 2.2.9

Known Issues

Boot Code

The kermit booter (`llkermit`) does not work.

Building Boots with the Wizard

The Wizard version 1.28 has an additional menu called **Customize**. The new **Choose Utility Subset** dialog allows you to select the utilities you would like and reduces the bootfile size. For backwards compatibility with the Wizard, version 1.27, you may use the full set of disk utilities and/or embedded ROM utilities. If you decide to use customized utilities, there will be no effect if you check the **ROM Utility Set** check box or **Disk Utilities** check box in the **Master Builder** dialog. Only selected utilities will be included. If you want to include the utilities by setting the **Disk Utilities** and/or **ROM Utility Set** flags, do not select any utility provided by the **Customize** menu.

Failure of Alarm Sequences

An error message was not generated when testing invalid sequences of alarms in user state and in system state:

```
A_ATIME(non-pending), A_DELET,
A_ATIME(non-pending), A_RESET.
```

In these two cases, an error message should have been generated because non-pending alarms (alarms that have already gone off) are removed from the system. Trying to delete a non-existent alarm should generate an error message, but the delete and reset commands did not return an error.

PCMCIA Interrupts

The SanDisk 4 MB and 20 MB PCMCIA ATA IDE cards do not work correctly with PCMCIA IDE interrupts. To make sure that all cards work with OS-9, the default mode for accessing PCMCIA IDE cards is polled mode.

For More Information

See the ***Enhanced OS-9 for Super H 7750SE01 Board Guide*** for information on enabling PCMCIA IDE interrupts.

Register Wait States

Due to the way the SH7750 operates its execution pipelines, CPU read-write access to some of the internal registers (such as SR - Status Register) and memory-mapped registers have to be followed by wait cycles. For example, performing a read instruction immediately after a write instruction on the SR does not return a correct result as one would expect. Therefore, it is necessary to place "dummy" wait instructions between the write and read instructions.

Since the number of wait cycles differs for various registers and that it may be changed in future variations of SH-4 processors, it was decided not to include the common register access routine in `cpu.l` and let the developers do the accesses as needed for their drivers. One of the software modules which had to take these wait states into account was `romcore`.

To provide future OEM users with flexibility to eliminate dummy wait states, the makefile for romcore (located at `\mwos\OS9000\SH4\PORTS\SH7750SE\ROM\ROMCORE\makefile`) has `MOD_LVECTORS` defined which controls the inclusion of wait states between writing and reading of the SR (status register). Disabling this flag takes these wait states off."

Selecting Java Support in Wizard

It is possible to select the checkbox labeled JAVA Support in the Configuration Wizard's Master Builder window even when the PersonalJava for OS-9 package is not installed. This checkbox should not be selected if you do not have PersonalJava for OS-9 installed. Selecting this checkbox for a build when PersonalJava for OS-9 is not installed will result in an error.

Additional Features

SCSI Support

Since the Solution Engine board had no onboard SCSI devices, there is no official SCSI support for this release. However SCSI drivers have been included in case OEM's want to add SCSI to their SH-4 hardware. Please note that these SCSI drivers have not been tested for SH-4 and are shipped "as is."

SH7750 Cache Mask

There is an init module definitions for the `m_cpucompat` field that will control cache behavior.

`CACHE_WT_MASK`: By setting this bit in the `m_cpucompat` field, the cache module will set up write-through mode for memory areas P0, U0, and P3.

By default, this bit is not set.

Supplemental Documentation

`iopt` Command Line Options

`iopt` has command line options to control processor-specific optimizations:

Table 5-1 iopt Command Line Options

`-e`	suppress all processor specific optimizations
`-ec`	suppress creation of CSEs containing complex constants
`-ef`	suppress creation of CSEs containing function addresses
`-eg`	suppress creation of CSEs containing global variable addresses

Each of these optimizations are enabled for SuperH processors. If the need should arise to disable them, use the options listed above.

Enhanced OS-9 for SH-3 Version 2.2

SH7709/SH7709A Solution Engine

The Ethernet driver used to be built from the /mwos/SRC/DPIO/SPF/DRVR/SP83902 directory. This has been changed. It is now built from the /mwos/SRC/DPIO/SPF/DRVR/SP83902_SH directory. The code has been enhanced for performance improvements.

EBX7709 Reference Platform

- There is no Super I/O (SIO) support for the parallel port.
- Keyboard support is available for serial ports SIO3 and SIO4.
- Mouse support is not available for serial ports SIO3, SIO4, and SIO5. These ports do not generate enough power to support a mouse This can affect Java applications that require a mouse.
- The EBX contains both a top and a bottom PCMCIA slot. However, due to hardware limitations, the top slot is limited to memory type devices (i.e. ATA flash cards), while the bottom slot is limited to I/O devices (i.e. Ethernet cards).
- If an Ethernet PCMCIA card is installed, software reset (i.e. “rst” from ROMBUG) will hang the system.
- Except for switch one, OS-9 does not use any of the DIP switches on the EBX board. The EBX hardware uses DIP switch 1 to determine which part of ROM to boot from.
- Currently, LCD support is not available; however, the video mode of the graphics controller is supported.

Configuration Wizard

This section describes the release notes on OS-9 for SuperH version 2.2.

- It is possible to select the checkbox labeled **JAVA Support Modules** in the Configuration Wizard's **Master Builder** window even when the **PersonalJava for OS-9** package is not installed. This checkbox should not be selected if you do not have **PersonalJava for OS-9** installed. Selecting this checkbox for a build when **PersonalJava for OS-9** is not installed will result in an error.
- The Configuration Wizard does not properly set the MAC address (or low level Ethernet address) in the SPF descriptors. No matter what the MAC address field in the wizard is set to, the SPF descriptors made by the Configuration Wizard will have this address set to 0. As a result, the SPF Ethernet drivers will always obtain this address from the hardware jumpers.

HawkEye

This section describes the release notes for HawkEye for SuperH, veresion 2.2.

- Hawkeye uses timer 0 on the SH7709 board.
- When starting HawkEye, select `File->Preferences->Target` and choose the **SuperH** processor type. HawkEye's current default processor is **PowerPC**.
- If the router daemon locks up, reset the target to capture any future profiles.
- The PC values corresponding to some of the kernel's internal system calls are not set to enhance performance.
- The process ID (PID) is set to the program that calls the kernel. Because of this, the system call event will show up in the log file as part of the calling process. Furthermore, the PC value is not properly set in the details window for the system call events.

- The stack indicated is the system stack. Some of the system calls include the following:

  `F_UACCT`

  `F_MOVE`

  `F_CMPNAM`

  `F_PRSNAM`

  `F_INITDATA`

  `F_SLINK`

  `F_TLINK`

  `I_IOEXIT`

  `I_IOFORK`

  `I_IODEL`

  `F_SALARM`

  `I_OPEN`

  `I_GS_DEVTYPE`

  `I_GS_DSIZE`

  `I_READ`

  `I_CLOSE`

  `I_GS_EOF`

  `I_GS_FD`

- If the filters are applied in the **Filter Criteria** dialog box, the system call events will not show up in the HawkEye application window. However, the details will show up in the **Details** dialog box.

Chapter 6: x86 Notes

This chapter provides an overview of the changes and improvements made to Enhanced OS-9 for x86.

Enhanced OS-9 for x86 Version 3.0

Threads Support Available

The latest version of the OS-9 kernel (version 125 or newer) supports multiple threads.

For More Information

See **OS-9 Operating System Notes** on page 151.

Compiler Changes

- The ROF format for 386 (x86) changed to support the threading specification. Old ones may not be compatible with new linker.
- The 386 (x86) linker will not accept ROFs or libraries created with a pre-Ultra C v2.4 assembler or libgen.

x86 Known Problems

CF4590: On a dual isa509 system, OS-9 may not use IRQ or IO address information programmed into the card using the 3COM utility.

The initialization routine of the spe509 driver for ISA cards pays no attention to the IO address and IRQ programmed into the card by the 3COM utility.

Enhanced OS-9 for x86 Edition 2

General Improvements

The following general improvements were made to Enhanced OS-9 for x86:

- Microware Hawk has been updated to use a new, faster protocol to communicate with the target processor. Quicker code tracing and animation sessions lead to faster application debugging sessions.
- Hawk more easily supports simultaneous debugging of multiple target applications. It is also possible to simultaneously debug two or more applications on different target hardware (even if they are different processors). This facility is particularly useful when the target systems communicate over a network.
- The TCP/IP and UDP/IP stacks are considerably faster.
- Generic compiler optimization improvements have benefited all processors.
- Java is updated to V3.0.1 (Java is an optional add-on package).
- The evaluation, OEM and BLS packages are integrated into one for the first time in this release. Hawkeye, Java, etc. are also integrated onto the CDROM as optional add-ons.
- CD-ROM reading ability is added to this package.
- The following new optional add-on packages are available:

  X.25/LAP-B WAN support

  TrueFFS flash support

  ISDN (Q.931)/LAP-D support

  SNMP

x86 Resolved Problems

- CF3038: x86 compiler: float -> unsigned short conversion is incorrect. Edition 43 or later of be386 fixes this problem.
- CF5228: ROF format cannot support many references. ROF series 9.1 was introduced for x86 and 68k targets to support greater than 65k references in a relocatable object format (ROF) file.
- CF6799: opt386 can get general protection fault in some situations. Edition 64 or later of opt386 fixes this problem.
- CF7316: opt386 should recognize and not fault on MMX instructions. The assembly optimizer did recognize MMX instructions. It also could not handle MMX instructions because it did not recognize the .q extension and did not recognize the mmx registers (%mm0 - %mm1) as valid registers. Edition 66 of opt386 has this support.
- CF7317: Enable MMX instruction disassembly in RomBug. Edition 15 Rombug has this support.
- CF7422: a386 does not recognize ffree instruction. The 386 assembler does not seem to recognize any of the following:

  ```
  ffree %st0
  ffree.x %st0
  ```

  The workaround was to do the following:

  ```
  #define FFREE(x) _asm(" dc.b 0xdd,0xc0+" #x)
  FFREE(0);
  ```

  Edition 62 of a386 fixes this problem.
- CF7452: FPU emulation is generating a bus error on 386/486SX/ELAN400. If one process is using the FPU emulation, the second FPU process can crash with a bus-fault at this position. When debugging a process that used the FPU emulation, the process may crash with a bus error also.

The problem was that the debugged process was not getting the appropriate accesses to the FPU. In fact, the debugger process was getting the access. This was also a problem where an FPU context could be allocated. Edition 7 of `fpu/fpuem` fixes this problem.

- CF7662: Incorrect instruction assembly. The assembler does not assemble the following instruction correctly:

  ```
  out.l %eax,#$32
  ```

  It is assembled as: e7 32000000, which is 3 bytes too long. The operand is only supposed to be 1 byte long.

  The problem with the `OUT` instruction being incorrectly assembled was a bug. Affected were the word and longword immediate versions of `OUT` and `IN`. This has been fixed. The fix will be available with Edition 63 or later of a386. The workaround is to assemble such instructions by hand (e.g., use `dc.b $e7,$32` instead of `out.w %ax,#$32`). Edition 63 of `a386` fixes this problem.

- CF7671: x86 RomBug disassembly crashes system.

  If RomBug tries to disassemble the following code, it will go into a seemingly endless loop, eventually taking down the system:

  ```
  out.b %al,#$32
  ```

  Edition 15 of RomBug should fix this. Fixed with latest disassembly code.

x86 Known Problems

The file `/mwos/SRC/DEFS/stream.h` is missing from the Enhanced OS-9 for x86 release. When the ISDN Communications Pak is installed, the libraries `isdnlib.l` and `isdnlib.il` can not be rebuilt.

Chapter 7: Processor Independent Notes

This chapter contains processor-independent release notes for OS-9 components. It includes the following sections:

- **ATM Notes**
- **OS-9 Compiler Notes**
- **Hawk Notes**
- **Hawkeye Notes**
- **PersonalJava Solution for OS-9 Notes**
- **SoftStax/LAN Communications Pak Notes**
- **1394 SDK for OS-9 Release Notes**
- **USB Peripheral SDK for OS-9 Release Notes**
- **MAUI Notes**
- **OS-9 Operating System Notes**
- **SNMP Notes**
- **OS-9 Utilities Notes**

ATM Notes

ATM 1.4 Notes

This section contains release notes for the Microware implementation of ATM (asynchronous transfer mode) networking software. It includes cumulative notes from versions 1.3 and 1.4.

Notes

An ATM device driver for the Motorola MPC8260 PowerQUICC II has been added in this release. This driver fully supports ATM Adaptation Layer 5 (AAL-5). ATM Adaptation Layer 0, OAM, and raw cells may be received and transmitted using the channel code 1 structures. The driver implements the Segment and End-to-End connectivity verification for OAM F5 flows at a User Network Interface (UNI) endpoint. The driver was developed using the Embedded Support Tools Corporation (EST) SBCATMF boards. The MPC8260 on these development boards used the following clock frequencies:

- External clock source (oscillator): 33Mhz
- Core: 200MHz
- CPM: 66MHz
- Baud-Rate Generator: 33MHz

The name of the driver module is `sp8260atm`. The names of its descriptors are `atm` (for AAL-5), `atm_aal0` and `atm_oam`.

Known Problems

A bug was seen in the PHY layer on the SBCATMF. Not all ATM cells arriving on the optical fiber are sent into the Rx FIFO in the PMC S/UNI-155-ULTRA PM5350. This problem appears to be unrelated to the ATM driver. The problem is seen when AAL-5 frames larger than 10 cells (480 bytes payload) are to be received.

MPC860SAR Modifications

Modifications to the ATM driver for the MPC860SAR were made. The driver was tuned so that it can transmit at 28Mb/s over one VC and receive at 39Mb/s over one VC. The driver was modified to receive AAL-5 frames larger than the mbufs allocated for receiving. These larger frames are now received into chains (linked lists) of mbufs.

ATM 1.3 Notes

This note is an addendum to ***Using the ATM Base Pak***, Chapter 4, Section ***Building the Boot ROM*** and to ***Using Soft-ATM™ for OS-9***, Chapter 2, Section ***Building the Boot ROM***. It adds details for building a boot ROM to be used for ATM.

When using the Configuration Wizard to build a boot ROM containing modules from the ATM Base Pak for a testbed using the 860SAR processor, you may wish to build a boot ROM that does not enable the Ethernet device on the MPC8XXFADS board, relying instead on IP over ATM.

Since the Configuration Wizard will automatically add Ethernet interface `enet0` to the IP interface table and routing entries for interface `enet0` to the IP routing table, the following instructions should be followed when building a boot ROM for ATM.

Step 1. Right click on the My Computer icon and select **Properties** from the menu. If there is a variable `MWMAKEOPTS` with value `CODO = codo.pl`, delete that variable and value.

Step 2. In the first Configuration Wizard window **select 860SARFADS** from the Port Selection window.

Step 3. In `Configure -> Coreboot -> Main Configuration`

tab **Debugger**	Select **RomBug** if the Ethernet is not used for debugging.

tab **Ethernet** Select an IP Address that has a LIS (Logical IP Subnet) prefix that is different from the LIS to be used to configure the ATM interface. If the Ethernet device is not to be used, select an LIS that will never be used by any software or users of the tested. Do not select an IP Gateway.

tab **Define Other Boot Options**
Select **Break** if the Ethernet port is not being used for debugging.

Step 4. In `Configure -> Bootfile -> Network Configuration`

tab **IP Address** This must be the same as the IP Address in tab Ethernet above. The LIS of the address must be different from the LIS used for the ATM interface.

tab **DNS Configuration** If the Ethernet device is not to be used, select **Disable DNS**.

tab **Gateway** Do not select a Gateway. The gateway will be added later with the route add default x.x.x.x command.

tab **SoftStax Setup** Select **Enable SoftStax**, **Start telnetd**, **Memory Pool Size 1024k**.

tab **SoftStax Options** Select **ifconfig**, **ndbmod**, **netstat**, **route**, **routed**, **telnet** if Soft-ATM™ is not used (no signalling). If signalling is to be used select **Soft-ATM** or **Soft-ATM Switch**.

Step 5. In `Configure -> Bootfile -> Disk Configuration`

tab **Init Options** Click on User then edit the Parameter List according to the instructions in Using the ATM Base Pak, chapter 4. If the Card Soft code is not needed for the PCMCIA device delete the command `csfd -z<>>>/nil&` from the

Parameter List. You may wish to use the more powerful mshell instead of shell as arguments to the `telnetd` and `ex` commands.

OS-9 Compiler Notes

Ultra C/C++ Version 2.4

Enhancements

- CF2326: _os_salarm_delete now attempts to default to the correct value for the pb.flags member. If ID 0 is specified, delete all alarms, it will default to TH_SPOWN. A new os_lib binding was created that allows the flag to be specified: _os_salarm_delete_sp.
- CF2530: xcc was enhanced to preserve I-code files specified on the xcc command line. Previous versions could corrupt or alter the I-code file which made it impossible to repeat the same command line.
- CF2608: cout was not properly printing strings that contained embedded '\n' characters. This has been fixed. For example, cout << "Hello\nWorld"; will print:
  ```
  Hello
  World
  ```
- CF6897: Including either rw/stdex/slist.h or rw/stdex/lhash.h would cause syntax errors. These errors were caused by a missing typename keyword on template type references. These errors were fixed by adding the typename C++ keyword in the appropriate places to these files.
- CF7516: xcc now correctly creates instantiation information (.ii) files when forward slashes are used in Windows path lists.
- CF7775: Assembly optimizers with instruction scheduling no longer incorrectly move code out of assembly language macros.
- CF7981: _os_ioconfig was corrected to take two parameters instead of one.
- CF8172: _gs_gfd now correctly fills the fields of the struct fd.

- CF8211: xcc now correctly handles options with optional parameters in the -z input file in C89 mode.
- CF8830: `tempnam` and `mktemp` are now able to produce 52 unique files before returning NULL. In addition, a new unique name is returned each time even if no files exist for previous unique names.
- CF9574: The default constructor for `RWTime()` no longer incorrectly calculates the current time.

Known Bugs

- CF9822 : If your MWOS environment variable is set with forward slashes, the C++ prelinker will complain with an internal error "`C++ prelinker: Input file not in list`". In addition, it will leave a temporary file called `_libgen_.tmp` in the current directory. Even if the environment variable is corrected, the internal error will be reported until the temporary file is deleted. This will be corrected in a future release of Ultra C/C++.
- CF10304: xcc's pre-processing options do not allow an equals sign to be used. When using `-pp`, for example, the destination directory is specified without an equals sign. This problem will be corrected in a future release of Ultra C/C++.
- CF10691: _gs_gfd for OS-9 doesn't properly take the passed size into account. To avoid this problem, ensure that a buffer of at least `sizeof(struct fd)` is allocated and pass `sizeof(struct fd)` as the size requested. This problem will be corrected in a future release of Ultra C/C++. _gs_gfd is an OS-9 for 68K compatibility function, use _os_gs_fd for new code.

Ultra C/C++ Version 2.3

This section contains release notes for the Microware compiler Ultra C/C++ version 2.3.

Problems Resolved

- CF2204: Program compiled with `-c` crashes with bus error. The following program ends with a bus error when compiled with the `-c` option. Edition 24 of `cpfe` fixes the problem.

```
#include <stdio.h>
typedef struct {
        const   char    *str;
} stru;
main()
{
        stru    data[] = {
               {"abc"},
               {"123"},
        };
        printf("c:%c\n", data[0].str[0]);
        exit(0);
}
```

- CF2373: `intercept()` and `signal()` do not work together. This problem has been corrected. The order of calling `intercept()` and `signal()` is irrelevant. Signals will be dispatched to the proper routine regardless of the order. The latest version of `clib` and `sys_clib` libraries fixes the problem.
- CF2397: Tables of function pointers in code do not work. Edition 24 of `cpfe` fixes the problem.
- CF2413 and CF2545: Long file names do not work using `readdir()` in `sys_clib.l. readdir()` has been updated to support PCF Long File Names. The latest `sys_clib` library fixes this problem.

- CF2428: `getwd()` does not work correctly for PCF. `getwd()` now uses `opendir()/readdir()/closedir()`. It was also modified to be non-recursive, to not `malloc/free` memory on each loop, and to not make an assumption as to the maximum path length to be returned. It will also abort if caught in a recursive loop, rather than crash like the predecessor. The latest `unix` library fixes this problem.
- CF2547: `getopt` should use `writeln` instead of `write.` When `getopt` prints an error message, it uses `write`. However, SCF does not process the `\n` as an end of line character, which causes problems in later messages. This should be changed to `writeln`. The use of `write()` to print error messages in `getopt()` was replaced with `_os_writeln(),` which comes from `os_lib.l`. Error messages should print correctly on SCF devices now. The latest `unix` library fixes this problem.
- CF2577: `strerror` should not cause problems with `errno.` Both `strerror()` and `perror()` avoid writing over the value in `errno` at function call time. `errno` is saved and restored so that it is constant across the call to `strerror()` and `perror()`. The latest `clib` library fixes this problem.
- CF2593: Add support for more time zones. Support has been added for a number of time zones and three new daylight savings time rules for Brazil, Australia, and New Zealand. Table 7-1 shows the added time zones.

Table 7-1 Time Zones Added

TZ	OFFSET	DZ RULE	NAME
MIT	660	-	Midway Islands Time
HST	600	-	Hawaii Standard Time
PNT	420	-	Phoenix Standard Time
IET	360	-	Indiana Eastern Standard Time

Table 7-1 Time Zones Added (continued)

TZ	OFFSET	DZ RULE	NAME
PRT	240	-	Puerto Rico Standard Time
CNT	210	usa	Canada Newfoundland Time
AGT	180	-	Argentina Standard Time
BET	180	brazil	Brazillian Standard Time
CAT	60	-	Central Africa
ECT	-60	european	European Central Time
ART	-120	-	Arabic Standard Time
EAT	-180	-	Eastern African Time
MET	-210	-	Middle Eastern Time
NET	-240	-	Near East Time
PLT	-300	-	Pakistan Lahore Time
IST	-330	-	India Standard Time
BST	-360	-	Bangladesh Standard Time
VST	-420	-	Vietnam Standard Time
CTT	-480	-	China Taiwan Standard Time
ACT	-570	australia	Australia Central Time
AET	-600	australia	Australian Eastern Time

Table 7-1 Time Zones Added (continued)

TZ	OFFSET	DZ RULE	NAME
SST	-660	-	Solomon Standard Time
NST	-720	new zealand	New Zealand Standard Time

Daylight savings time rules are as follows:

usa: begins first Sun in April, ends last Sun in Oct

eur: begins first Sun in March, ends first Sun in Oct

bra: begins first Sun in Oct, ends first Sun after Feb 11

aus: begins first Sun in Oct, ends last Sun on March

nz: begins first Sun in Oct, ends first Sun after March 15

The latest clib library contains the updates.

- CF2649 and CF5443: Additional version information requested for the compiler. The compiler will now show its full version information if a `-tp=<target>` option is used and no source files are given. For example:

```
xcc -tp=ppc
Microware Ultra C/C++ Compiler. Version 2.3
Copyright 1999 Microware
PC version
xcc        Edition  67, Nov 10 1999
cpfe       Edition  32, Nov 10 1999
ilink      Edition  11, Nov 10 1999
iopt       Edition  97, Nov 10 1999
ficode     Edition   2, Nov 10 1999
beppc      Edition  59, Nov 10 1999
optppc     Edition  65, Nov 10 1999
appc       Edition  56, Nov 10 1999
prelink    Edition   9, Nov 10 1999
linker     Edition 151, Nov 10 1999
```

This is supported with version 2.3 or later of Ultra C/C++.

- CF2743: `cpfe` internal error. The following code results in a `cpfe` internal error when compiled:

  ```
  /* foo.c */
  main()
  {
         char    *a;
         int     *b;
         int     n;
         n = a - b;
         /* "n = a - (char *)b;" is OK */
  }
  ```

  This has been fixed in `cpfe`, Edition 30 or later.

- CF2931: C-library function bus error (102) when there is insufficient memory. This error can be duplicated on any OS-9 machine by starting a shell with most of the memory and then by running any utility which prints out data. Some amount of trial is necessary depending on the system. This error case can happen on any C FILE type access on this error path. The latest release of the `clib` library fixes this problem.

- CF2954: `use` directive defaults to non default directories. All assemblers (`os9` and `os9k`) will default to looking in `/dd/defs` for files included using the `use` directive in the absence of any other suggestions (for example the `-u=<directory>` command-line switch or the CDEFS environment variable). This is not a good solution on cross-compilation systems (for example Windows).

  The default `use` directive search directory, in absence of a given search path (via the DEF `envvar` or the `-u` switches) is as follows:

  1) For resident systems the default `dir` remains `/dd/defs`.

  2) For cross-hosted systems the default dir is `/mwos/<OS>/SRC/DEFS` where <OS> is OS-9 for 68K and OS-9000 for all others. This was fixed in the following editions of the assemblers:

  r68: 203

  a386: 60

  others: 53

- CF2955: `cpfe` converts calls to functions as calls via a function pointer. In the absence of a function prototype, `cpfe` is transforming calls to a function as if they were called via a function pointer. This was fixed in `cpfe`, as early as Edition 28.
- CF2968: Unnamed bit field, initialization causes internal error. The following program causes `cpfe` to terminate with an internal error:

  ```
  struct bob {
          unsigned short a;
          unsigned int :16;    /* note: un-named */
          unsigned long b[32];
  } a_bob = { 1, {1, 2} };
  ```

  This was corrected in Edition 23 of `cpfe`.
- CF2997: Assembly optimization: location tracking misses some gratuitous moves. The location tracking transformation does not process the following very well (code example is from PowerPC):

  ```
  nop
  li r2,15
  mr r1,r2
  mr r2,r1
  nop
  ```

  The second `mr` could be eliminated but it is not (actually, copy propagation will eliminate it, so this should be done with copy propagation disabled). However, if the `li` is replaced with a memory load, the operation completes:

  ```
  nop
  lwz r2,0(r3)
  mr r1,r2
  mr r2,r1
  nop
  ```

  The second `mr` is eliminated.

  This has been fixed in Edition 63 or later of the assembly optimizers (`opt68k`, `opt386`, `optppc`, `optarm`, `optsh`, `optmips`, and `optsparc`).

- CF3039: GPF in DOS version of linker. The linker may terminate after emitting "value too large for field"-type errors. This was fixed in Edition 149 or later of the linker.
- CF3055: The `is*()` (for example `isascii()`) functions in `ctype.h` should process large `ints.`

  The `is*()` routines in `ctype.h` do not operate well with values outside the range of an unsigned `char`.

  The `is*()` functions declared in `<ctype.h>` have been modified to check for `invalid` input (values that are not representable as an unsigned `char` and not equal to the value of the macro `EOF`) and return 0 for such input.

  The macros defined in `<ctype.h>` do not have such safeguards. As a result, they are faster. Since the macros are the default, to use the functions you must explicitly use the `undef` preprocessor directive or ensure that the macro expansion does not occur (by placing the function name within parentheses). The latest release of the `clib` library contains this update.
- CF4853: RFI- Place compiler error at the end of verbose output. In Hawk, the verbose output and error messages from various phases seem to be incorrectly intermingled. Edition 66 or later of `xcc` fixes this problem.
- CF5242: Time setting contains many errors. If the `TZ` environment variable is not set, the minutes west of Greenwich Mean Time is taken directly from the system's `init` module. Previously, if the value from the `init` module did not match a known time zone no adjustment was done for time zone by functions such a `localtime`.
- CF5563: `scanf()` function handles `%n` incorrectly in some cases. The `%n` conversion specifier doesn't work if you have an exact format match (in other words, if you hit `EOF` on your input). The latest release of the `clib` library fixes this problem.
- CF5586: C libraries have `internal iopt` warnings. When `-cw` (get verbose warnings) is used, certain functions in the C libraries may give warnings which can not be controlled by the programmer. The latest release of the compiler libraries fixes this problem.

- CF5662: `clib.l` - LIBM version of math routines incorrectly generates exceptions. Certain math functions might generate exceptions given certain error conditions. For example:

  `pow(0.0, -1)`

  The ANSI standard is as follows:

  `<math.h>`

  4.5.1 Treatment of Error Conditions

  The behavior of each of these functions is defined for all representable values of it input arguments. Each function shall execute as if it were a single operation, without generating any externally visible exceptions. This allows exceptions to occur, but they must be handled by default. The latest release of the compiler libraries fixes this problem.

- CF5693: `iopt` support for processor-specific optimizations. The compiler performs common subexpression transformations on global addresses, most floating point constants, and large immediates (not representable with two instructions) when targeting ARM/StrongARM. When targeting PowerPC, SPARC, and MIPS CSE transformations are done on floating point constants (as they must be loaded from memory), and large immediates not representable in one instruction. Edition 93 `iopt` supports the `CSEfying` optimization for ARM, PowerPC, SPARC, and MIPS.

- CF5719: `iopt` can incorrectly move the wrong code during common code motion. It's possible for `iopt` to incorrectly move statements around when doing common code motion optimization. For example:

  ```
  irq_disable();
  do {
      check_something();
      irq_disable();
  } while (cproc == 0);
  ```

  can get changed to:

  ```
  irq_disable();
  do {
      check_something();
  } while (cproc == 0);
  ```

When what it meant to do was:

```
    goto inloop;
    toploop:
        check_something();
    inloop:
        irq_disable();
        if (cproc == 0)
            goto toploop;
```

Disabling common code motion with `iopt`'s `-h` option works around the problem. Other factors may also hide it as well. Edition 89 or later of `iopt` fixes this problem.

- CF5790: `iopt` terminates abnormally on some functions. Certain (largely dead code) functions can cause `iopt` to GPF on Windows. For example, this source file causes a crash:

```
void func(void)
{
    extern int glob;
    switch (glob) {
        case 1:
        case 2:
        case 3:
            break;
    }
    return;
}
```

Edition 90 or higher of `iopt` fixes this problem.

- CF5869: `cpfe` - creates bad implicit prototypes for functions defined `in _asm()`. Given the following code (`ick.c`):

```
#ifndef NOASM
_asm("function: ");
#endif
int func(int x, int y)
{
    char buf[64];
    if (function(x, buf))
        return x + y;
    return 0;
}
```

Compiled with the following (the target is irrelevant):

```
xcc -tp=68k -efe ick.c -cw -fd=ick1.i
xcc -tp=68k -efe ick.c -cw -dNOASM -fd=ick2.i
```

The front-end is looking through the `_asm()` (in the first example) for labels that appear to be function definitions (labels that start with an `isalpha()` char and contain only `isalnum()` chars and end with a ':' -- this misses those with '`_`' and will also catch variable definitions) and generating what amounts to a prototype for these "functions".

Fixed in edition 28 of `cpfe.exe`. No longer attempts to track "functions" defined in `_asm()` statements.

- CF5892: `xcc` - allow overriding of target suboptions with default suboptions. The compiler executive allows you to override previous options with new ones (when reading left to right). The following example demonstrates this ability:

```
> xcc -bh x.a -eas -tp=68k -tp=ppc
```

Include file paths:

```
 \mwos\SRC\DEFS
 \mwos\OS9000\SRC\DEFS
 \mwos\OS9000\PPC\DEFS
appc -a=_UCC -a=_MAJOR_REV=2 -a=_MINOR_REV=2 -a=_SPACE_FACTOR=1
-a=_TIME_FACTOR=1 -a=_OS9000 -a=_MPFPOWERPC -a=_FPFPOWERPC -a=_BIG_END
-o=x.r -q -u=\mwos\SRC\DEFS -u=\mwos\OS9000\SRC\DEFS
-u=\mwos\OS9000\PPC\DEFS x.a
```

Note that the target is PowerPC, not 68k.

However, this ability was not provided for the target suboptions.

For example:

```
> xcc -bh -xio -ebe x.i -tp=armv4,vld -tp=ppc
```

Include file paths:

```
        \mwos\SRC\DEFS
        \mwos\OS9000\SRC\DEFS
        \mwos\OS9000\PPC\DEFS
    beppc -o=x.o -px -m=0k x.i
```

Note that the target changed (`beppc` is used instead of `bearm`), however the `bearm` option `-px` (which corresponds to ARMV4's ,`vld`) is still present. `beppc` cannot function with this option.

You can determine the default target suboptions by asking for them explicitly. Note, however, that if you use `-tp=ppc,vld` the executive merely ignores the suboption (because it doesn't exist for PowerPC), but keeps the `-px` on the command line.

Note that this handicap does not seem to infect the assembler phase:

```
> xcc -bh -eas x.a -tp=68k,sc -tp=sparc
```

Include file paths:

```
 \mwos\SRC\DEFS
 \mwos\OS9000\SRC\DEFS
 \mwos\OS9000\SPARC\DEFS
asparc -a=_UCC -a=_MAJOR_REV=2 -a=_MINOR_REV=2 -a=_SPACE_FACTOR=1
-a=_TIME_FACTOR=1 -a=_OS9000 -a=_MPFSPARC -a=_FPFSPARC -a=_BIG_END
-o=x.r -q -u=\mwos\SRC\DEFS -u=\mwos\OS9000\SRC\DEFS
-u=\mwos\OS9000\SPARC\DEFS x.a
```

Note that the `-j` switch is not passed to `asparc` (it would be if the target were still 68k). The compiler now correctly resets target suboptions to their defaults. Edition 64 or later of `xcc` fixes this problem.

- CF6053: `iopt` can corrupt semantics of structure copy. `iopt` does not seem to acknowledge that when a composite is copied and one version of the composite is modified, it should keep both copies as the user wrote. The code that demonstrates this is as follows:

```
int glob;
typedef struct {
    int lX, lY;
}       int_2d_t;
typedef struct {
    int_2d_t    stOrg, stDim;
}       im_win_t;
int
lCopyStruct(im_win_t stWinProcess)
{
    im_win_t    stWinCpy;
    glob += stWinProcess.stDim.lY;
    stWinCpy = stWinProcess;
    stWinCpy.stDim.lY /= 2;
    glob += stWinCpy.stDim.lY;
    glob += stWinProcess.stDim.lY;
    return 0;
```

`iopt` uses `stWinProcess` for every composite reference shown. Edition 91 or later of `iopt` fixes this problem.

- CF6094: #ident causes front-end to terminate with internal error. If a source file contains a #ident directive and is compiled in ANSI extended mode, an internal error "At end of source: internal error: C2Istmt.c:SetPragmas: no pragmas!" is generated. In ANSI mode this is correct. The following program demonstrates the problem:

  ```
  #ident "This is who I am"
  int f(int p) { return p; }
  ```

  Edition 32 or later of cpfe correctly flags this as an illegal preprocessing directive.

- CF6099: rename() memory leak. The OS-9 rename() function in the clib library (on non-68K targets) has been updated to correctly free allocated buffers upon successful completion. The latest release of the clib library fixes this problem.
- CF6238: xcc's new preprocessing options do not function correctly. The implementation was changed to correctly parse the file name and allow the mode to be in lower case. In addition, the -P option in C89 mode now can be followed by a mode option to get similar functionality. The C89 modes are upper case. Edition 65 or later of xcc fixes this problem.
- CF6242: Preprocessing to files does not function properly, it always goes to stdout. Edition 31 or later of cpfe fixes this problem.
- CF6606: iopt leaves code with extraneous pointers. For example:

```
static void func(int *a0, int *a1, int x)
{
        if (x)
        {
                *a0 = 1;
                *a1 = 2;
        }
        else {
                *a0 = *a1 = 0;
        }
}
int callit(int a, int x)
{
        int a0, a1;
        func(&a0, &a1, x);
        return a + a0 + a1;
}
```

iopt will leave all the pointers in where it should have deleted the pointer symbols and referred to the symbols directly. Edition 95 or later of iopt fixes this problem.

- CF6977: iopt code motion hoists function calls above global references. If the function call hoisted modifies the global in question, the generated code will not function properly. Edition 96 or later of iopt fixes this problem. Edition 97 improves this fix.
- CF7026: iopt can terminate with a segmentation fault if enough optimizations are suppressed. If CSE is not suppressed and other optimizations are, iopt can result in a segmentation fault. An example of such a command line is as follows:

  ```
  iopt -o -he -cd -cl -ct <file>.i
  ```

  Edition 98 or later of iopt fixes this problem.
- CF7515: Assembler failed to compile files containing "-" (hyphen character). The problem has been fixed. It is available in the following editions of the assemblers:

  r68 - Edition 205

  a386 - Edition 64

  aarm, appc, ash, amips, asparc, etc. - Edition 57.

Hawk Notes

This section contains release notes for the Microware Hawk Integrated Development Environment. It includes cumulative notes from versions 1.2, 2.0, 2.1, and 2.2.

Hawk 2.2

Enhancements

- CF3999: Hawk: ability to run other applications within Hawk. It is now possible to insert a batch file into the project which can be used to run utilities, like os9merge, after the project is compiled.
- CF7190: Hawk improvements. The order that components are now compiled is the same. They are compiled from the component listed first in the project window down to the bottom one with a child component being compiled before the parent.
- CF7821: Focus for function keys is lost in debug mode. The function keys that are used to do various functions like step and next in debug mode were working only if the focus was on the source window. Now the function keys will work when the focus is on any of the other debugging windows, like the process I/O window, as long as the window is docked.
- CF8947: Access violation & memory freed twice on close project space. On some occasions when exiting Hawk, Hawk tried to destroy objects that were no longer present causing access violation and memory freed twice errors to occur for each nonexistent object. Hawk has been fixed to check to make sure an object actually exists before attempting to destroy it.
- CF9008: RFI: Increase size of Hawk Debugger Process I/O Window. The size of the Hawk debugger process I/O window buffer was increased to 400 lines.

- CF9064: Problem with src conformance property settings in Hawk V2.0. CF10132: error in project - component properties. Every component in Hawk used to have the same source conformance settings as the project and changing the value for one component would change the value for every component in the project. Hawk has been updated to allow each component to have its own setting.
- CF9103: In StrongArm Hawk/Hawkdbgr, Step and Next fail in function preamble. When a branch and link is done in the function preamble, a breakpoint is set on the instruction right after the branch and link, but that instruction is never executed. As a result, the program would run until it exited. The dll `mwsrcdbg.dll` has been updated to notice that condition and take appropriate action.
- CF9417: Saving of environment variables for the target. Hawk had not been correctly recognizing newline characters when it read in environment variables that were saved in the project files. Version 2.2 of Hawk now recognizes the newline character correctly and restores saved environment variables correctly.
- CF9421; RFI: Handle carriage returns in the define/undefine boxes. Carriage returns can now be used to separate items when they are entered in the define/undefine boxes in the project's options.
- CF9487: Can not delete environment settings through Debug -> Options menu. Environment settings which had been deleted were restored the next time Hawk was opened because they were not properly deleted from the registry. In the environment options dialog box, the delete button now deletes the item from both the dialog box and the registry so it will not be restored the next time Hawk is opened.
- CF9552: Hawk debugging problems - watches not evaluating variables. Previously, the expression `(abc + 0)->x` would not be evaluated correctly if it was inserted into the watch window. There had been a problem in `mwsrcdbg.dll` which caused it to interpret addition with a pointer incorrectly. This has been corrected and the previous expression is evaluated correctly now.

- CF9785: Problem with Debugging. There was a hard-coded size limit on the length of a symbol name in a `.stb` or `.dgb` file. Occasionally, a symbol was generated that was too large which would cause an access violation when it was read. Hawk now automatically increases the size limit when a symbol which previously would have been too large is read.
- CF9795: Hawk Process I/O emulation not fully vt100. Hawk's process I/O window has been updated so that it supports VT100 terminal escape sequence commands.
- CF10010: Hawk Enhancements. A run to cursor option was added to the assembly window's pop-up menu.
- CF10082: No warning that LibPreLoad key is removed. Hawk now uses the MWOS environment variable setting to locate the .dll's that it needs instead of using the LibPreLoad registry key.
- CF10123: Question about Hawk Version 1.2. Every time that a Hawk project was saved, the description in the component description box was copied and appended to the end of itself which would result in the same description being repeated over and over again in the component description text box. This has been fixed so it no longer happens in version 2.2 of Hawk.
- CF10130: Hawk Component output names can no longer contain `.exe`, `.lib`, or `.so`. If a user uses an extension to an output name like `.exe` or `.lib` Hawk would delete the extension and assigns its own filename extension. Version 2.2 of Hawk no longer does this.
- CF10183: Hawk doesn't use CSL. If "Shared" was selected as the link type in the project properties, the command line that was created did not always correctly include the options to use the CSL. This has been fixed in version 2.2 of Hawk.
- CF10234: "UK24569 Hawk 2.1 68k Build Failure" When checking to see if a file had been modified, Hawk did not always look in the correct directory. When Hawk examined the wrong directory and could not find the file, Hawk would incorrectly return an "Up to date" message when a user attempted to build their project. Hawk has been corrected to locate a project's source files correctly which fixes this problem.

- CF10353: Combo Box in the Menu Target -> Load doesn't work correctly" Only one item in the Target->Load combo box was being saved because the box had been set up to save its text and not its items. This has been fixed and the combo box will now save its previous 8 entries.
- CF10469,"Hawk doesn't seem to work on 405GP" The 405GP is a processor that was not previously supported, so Hawk had to be updated to know about it and its register set. Hawk has been updated and it will work with a 405GP processor as its target.

Hawk 2.1

XML Project Format

The Hawk project file (`*.mpj`) format has been changed from the .ini file format to the XML file format. The conversion of existing projects will be seamless. An editor is provide with Hawk 2.1 for editing the project files manually.

WARNING

Opening an old Hawk project in the new version of Hawk (version 2.1) will make that project incompatible with previous versions of Hawk because Hawk will covert the project to XML format.

Hawk Profiler Improvements

The following improvements were made to the Hawk Profiler:

- Addition of Memory Usage, and CPU Usage chart.
- Completely re-written and easier to use. Some features that were added include selecting targets, saving profile data to a text file, and displaying function data for multiple modules at the same time.

- The profile data is now stored in a table format. This fixed column alignment problems with the previous profiler.
- The Hawk Profiler was rewritten as a Java application. To use the Profiler, you must have a Java Runtime Environment (JRE) installed on your system. The Hawk Profiler was tested against the Sun JRE version 1.2.2, which is provided on this product CD.

General Improvements

- CF9850: Hawk: Trying to add a previously unsaved file to a project results in a crash. This was fixed in version 2.1 of Hawk.
- Hawk now looks in the `\MWOS\dos\bin` directory (the MWOS location is read from the system's environment variable settings) for `hawkdbg.dll` and `hawkprj.dll` instead of looking at the `libpreload` registry keys as in previous versions.
- A button for the Profiler has been added to a toolbar on the project panel.
- CF2521: Support for the “Go with Boot Staging” command, gb, has been added to the Windows debugger. The RomBug command gb has been added to Hawk.
- CF4176: Previously, aborting a system state connection did not work. Aborting a system state connection using the `STOP` option when the debugger is not active was not supported. Currently, Hawk can handle the abort of a system state connection.
- CF4178: Previously, it was not possible to view properties of the module and **OTHER** from the symbol browser. Hawk would crash if the user selected to view the properties of the module or **OTHER**.

  This crash no longer happens, and if the properties of the module are selected, Hawk completes an ident of that module on the target system and displays the information. The **ident** feature has also been added to the target drop-down menu.

- CF4190: A **Clean all** option has been added to Hawk. It is accessed through the project's pop-up menu. Currently, the incremental link option is not available with the compiler; it will be added to Hawk at a later date.
- CF4276: Previously, there were problems debugging modules of different files with the same source file name. This problem has been corrected; Hawk has been changed to use the full path names when searching for files.
- CF4297: A **Load all** option has been added to Hawk. It is accessed through the project's pop-up menu. When selected, all of the modules in a project are loaded to the target specified in the module's properties.
- CF4948: Previously, Hawk did not support building of C++ libraries containing templates. This problem has been resolved in Hawk by doing a prelink for both I-Code and O-Code C++ libraries.
- CF5137: Previously, Hawk did not support moving a project. This has been resolved; a dialog has been added that is called when Hawk detects a project has moved. The dialog gives the user two options to resolve relative sources. The options are listed below:

  Assume the sources remain in their original location.

  Assume the sources were moved relative to the new project location.
- CF6248: Previously, Hawk would cause the PC to hang when multiple `PreLoadLib` dlls were specified. Currently, Hawk can handle this task provided there is more than one set of registry entries for `PreLoadLib`.
- CF6924, CF7007: The Help file `HAWKWT_10.HLP` appears to be missing from the products. The Help system no longer requires this file.
- CF6985: Hawk 1.2 for 68k is installing old and wrong `cpu.l` for 68020 Product. An unusable version of `cpu.l` was placed in the `MWOS/OS9/68020/LIB` directory on Enhanced OS-9 for 68K V1.0 distributions. The file has been removed from the distribution.

Note

If the file `cpu.l` exists in your `MWOS/OS9/68020/LIB` directory, it should be deleted.

- CF7009: Hawk libraries not updated.
- CF7242, CF7276: Enhanced OS-9 for 68K Evaluation Installation will not complete. The `dbgextns` module was incorrectly left out of the distribution. The module must be initialized via the init module's extension list or via the `p2init` utility prior to using Hawk.
- CF7397: Previously, Hawk's New Project/Unit dialog had a bad default button. The default button selected in the final stage of creating a project was the **Next** button, which was not enabled in that particular dialog. The default button has been changed to be the **Finish** button.
- CF7414: Previously, Hawk batch file units did not run properly on Windows 95. Hawk now creates the command line using the `COMSPEC` environment variable. When the full path of `command.com` on 95 and 98 is used, the batch file executes correctly and captures the I/O.
- CF7421: Previously, Hawk's project builder passed the unsupported option `-bepg` to `be68k`. The 68k back end has been modified for this release so that it now accepts the `-pg` option.
- CF7514: Previously, it was impossible to build single source modules with Hawk. Now there is an option to exclude a unit from the link. In the **Properties** window for any given unit, you should see a check box with the caption **Exclude from Link**. If this box is checked, the unit is still treated as a dependency of the component, but it will not appear on the command line when that component is linked (or a `libgen` is performed).

  In addition, a change was made to add an option that associates a unit as the "Unit Root Psect" of a component. This is an easy way to add a unit to a component and then specify that the added unit is the

Root Psect. In the **Properties** window for a component, there is a combo box called "Unit Root Psect". It contains a list of the units in that component. Select one of these to be your Root Psect. Note that this option takes precedence over the Root Psect option on the **Link** tab of the **Properties** window.

- CF7639: Hawk turns off all optimization by default for components. When a new component is created, the properties are synchronized with that of its parent. The parent may either be another component (if the new component is a child), or the project (if the new component resides at the root). After the properties are synchronized with the parent, the properties in `HAWKDEFS.INI` are applied based on component type. `HAWKDEFS.INI` resides in the Windows directory and can be freely modified by the user.
- CF7677 and CF7679: When changing the stack size, Hawk previously used increments that were too large. For this release, Hawk has been changed to use more reasonable units.
- CF7753 and CF7047: Previously, Hawk had overwritten settings for `*.c`, `*.cpp`, `*.h` extensions without warning in older versions. Instead of overwriting current settings, the current version now asks you if you would like to overwrite their current settings.
- CF7822: Previously there was no help available in `Debug->Memdisplay`. In the current version of Hawk, if you click **Help** from the memory window, it displays context sensitive help.
- CF7823: Previously, a Hawk help file system failure occurred. In the current version of Hawk, when `Help --> Index---> Contents---Debugging--->Attach to process` is selected, error 129 no longer appears.
- CF7859: Previously, it was not documented that the target system must have the module’s pipe and pipeman in memory to use the Hawk Debugger. This has been added to the current documentation.
- CF7904: Previously, the Hawk help files were shipped "As Is" from Premia Corporation and documented the Codewright 6.0 editor only. The non-Hawk-related functionality has been removed from the documentation in this release.

- CF7907: Previously, the **Add existing project** dialog in the file view window called the incorrect function. Currently, the menu calls the correct function.
- CF8071: Previously, the Hawk Assembler GUI displayed only assembly listed in the **show address** mode. Currently, the default for this window displays mixed source in **show function name** mode.
- Resolved issue number 8109: The following `dbgextns` modules were missing from the Enhanced OS-9 for 68K Version 1.0.

  Modules: `/MWOS/OS9/68000/CMDS/BOOTOBJS/dbgextns`
  `/MWOS/OS9/68020/CMDS/BOOTOBJS/dbgextns`

  The `dbgextns` modules were not included on the Enhanced OS-9 for 68k Version 1.0 Distribution. These files are now included in this release.
- CF8118: Previously, if Hawk exited when a project was not open, it tried to destroy unopened windows which resulted in multiple `memory block freed twice` errors. The new version of Hawk checks to make sure the window exists before it tries to destroy it.
- CF8122: Component names that began with an underscore character were not displayed in the project window. This has been fixed. The current release of Hawk allows a component name to be up to 48 characters. The component name may contain any characters. The output name of the component is validated to allow alpha-numeric, '`.`', '`_`', or '$' characters. Any invalid characters are replaced with an underscore.
- CF8125: There were several problems with Hawk's help system. The help system has been updated to be current, and all of Hawk's menus and indexes access the correct files.
- CF8147: If you were debugging code and pressed **step** on the last line of source code, error (106:006) would be displayed. `Mwsrcdbg.dll` has been modified so that it exits correctly without reporting an error when a step is done while on the last line of source code in a program.

- CF8188: Hawk only allowed the user to input numbers when editing registers. As a result, the value of a floating point number could be changed only to an integer. The new version of Hawk allows users to enter numbers and decimal points in the floating point registers window.
- CF8191: Previously, Hawk required the user to hit **Enter** twice after changing the value of a variable in the locals or watch window. For this release, these windows have been modified so that the user only needs to hit **Enter** once.
- CF8287: On the previous **Connect** dialog, if you chose to attach to a module, a **Browse** button was made available to select the appropriate module. This button was removed because the module to be entered in the box was intended to be the one currently loaded on the target. Entering a module name with a full path was never intended and consequently did not work.
- CF8413: Previously, when you selected `Hawk -> Help -> Configure Index File`, files were listed that were not shipped. The index (`mwhawk.idx`) has been updated in this release to include the proper files.
- CF8548: Upon running Hawk for the first time, an error dialog would display with the following message, "System or Project Initialization Failed." This occurred because the `mwhawk.ini` file that shipped had "ClearCase" listed in the VCS entries. This was removed from `mwhawk.ini` in the current release.
- CF8566: `mwhawk.tdx` is the index file used by the Hawk Assist feature. It referenced files that did not ship with the product. The index file has been updated.
- CF8670: Previously, if Hawk failed when you attempted to load a module, no error message was displayed. So it would appear to the user as if the module loaded correctly even though the load failed. In this release, if Hawk fails to load the module or fails for another reason, an error message is displayed.
- CF8844: Previously, the **Group** and **User** edit boxes only allowed for three digits of input. Currently, five digits can be input; this allows for numbers up to 65,535.

- CF8849: Previously, Hawk crashed if you pressed **Build/Rebuild** without having a project space open. This problem has been fixed for the current release.
- CF8991: Previously there was an error in the Hawk **Component** menu. When a project space was opened, and you right-clicked a unit, the **Component** menu would pop up. If **Remove from project** was selected using this menu, the whole component would be removed. In the current release, this has been corrected.
- CF9011: Previously, SoftStax and MAUI functions were not available with the Hawk Assist feature. These are now available.

On-Line Help System

The on-line help system was improved to reflect an upgraded version of the code editor and new compiler and debugger functionality. The following sections describe the on-line help system.

System Requirements

The OS-9 on-line help system includes both of the following types of help files. The requirements for each are also listed.

- Standard WinHelp 4 (`.hlp`) files. To view, these require Windows 95 or later and the `RoboEx32.dll` file (this file is shipped and installed with Enhanced OS-9 products).
- Microsoft Compiled HTML Help (`.chm`) files. To view, these require Windows 98 or Windows 2000. Windows 95 and Windows NT users must have Internet Explorer 4.0 (Service Pack 2). This is available for free from the Microsoft web site.

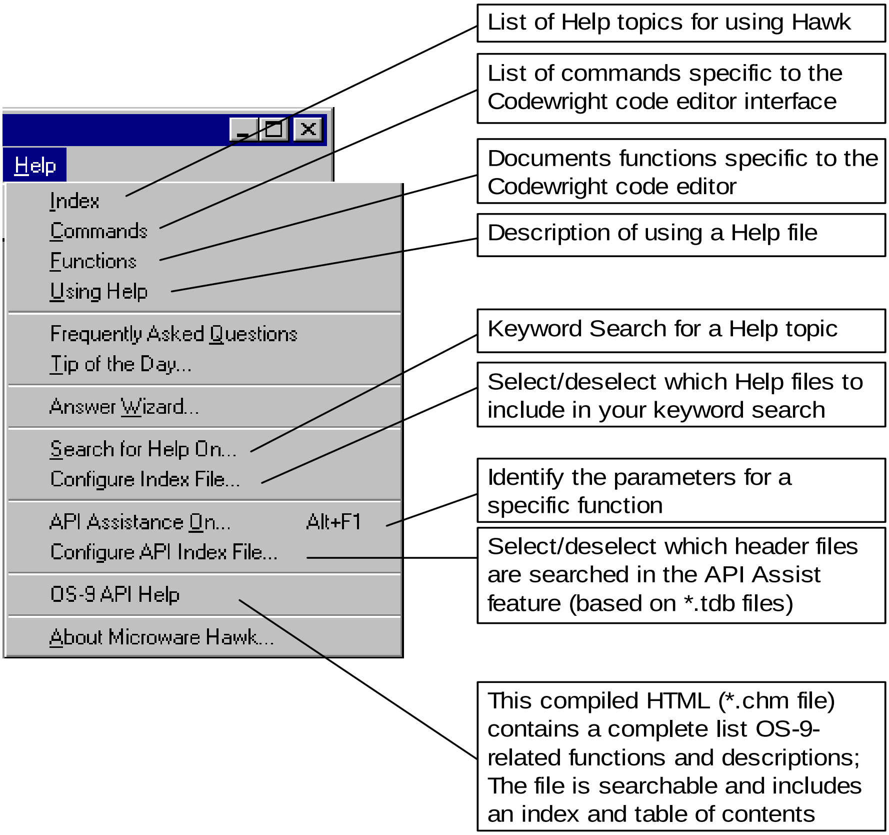
Note

All information provided in `.chm` format is also provided in PDF format.

Using the Help System

The OS-9 Help system is accessible from the Hawk integrated development environment user interface. **Figure 7-1** shows the Hawk Help menu and the Help options it contains.

Figure 7-1 Hawk Help Menu



List of Help topics for using Hawk

List of commands specific to the Codewright code editor interface

Documents functions specific to the Codewright code editor

Description of using a Help file

Keyword Search for a Help topic

Select/deselect which Help files to include in your keyword search

Identify the parameters for a specific function

Select/deselect which header files are searched in the API Assist feature (based on *.tdb files)

This compiled HTML (*.chm file) contains a complete list OS-9-related functions and descriptions; The file is searchable and includes an index and table of contents

In addition to the Help menu, most dialog boxes in Hawk will include a Help button. These are usually located in the lower right-hand corner of the dialog box. Selecting the Help button automatically selects the Help topic associated with the particular dialog box.

The F1 key no longer accesses the OS-9 API help topics because they have been converted to `.chm` format.

Hawk 2.0

For More Information

For complete information on how to use Hawk, refer to the following documentation.

Getting Started With Hawk

Using Hawk

Using Hawk Macros

The Hawk On-line Help System (accessed from the Hawk Integrated Development Environment).

Editor Enhancements

- CodeSense

  CodeSense provides Word Completion and function parameter help. Graphical symbols distinguish parts of code, and 'Goto buttons' take you to their definition.

- Project Spaces

  Our new Project Spaces allow you to view multiple projects at one time without physically changing which project is the "current" project.

- CodeFolio

  Do you have pieces of code you or your team use frequently? Work faster by clicking and dragging your most commonly used code stored in a folder or file view. This enables not only a quick placement, but also macro expansion and prompted input as you drop.

- Multiple Clipboards

  Multiple clipboards and scrap buffers give you added room for storing code segments for use later in your editing sessions.

- Command Line Editing

  Enhanced command line editing allows greater flexibility in inserting text on the command line.

- Clip View

  View the contents of your clipboards or scrap buffers.

  Updated Language Dialog

  Associate View Setups or Lexers to your file type for added editing power.

- ChromaCoding Lexer

  Augment or change the support for an existing language. Lexers allow you to define comments, keywords, numbers and strings for your language without having to write a DLL.

- Window Manager

  The Window Manager dialog allows you to selectively tile or set display attributes for individual windows or documents.

- View Setups

  View Setups both consolidate and expand the most common customizations into a single dialog. Control color effect, font usage, outline effects and column wrapping.

- API Macros

  Take our interactive commands from the Hawk API: add loops, block, if...else..., variables, parameters and comments, and you have a C-like macro language that is quick and handy.

- Enhanced Word Wrap

  Hawk provides you with true soft word wrapping. Word wrap without reformatting your code.

- Auto-hide Toolbars

  Give yourself more editing space by using auto-hide toolbars.
- Customize Menu

  All of Hawk's customization features are located in the **Tools**-> **Customize** menu.
- Enhanced Auto-Save

  Additional file specification options as well as the ability to limit the size of files that are saved.
- Popup Menu Editor

  The new Popup Menu Editor may be accessed by selecting the "Edit this menu" option on most Hawk popup menus.
- Hex Mode Editing

  Expanded hex editing features include copy, cut, and paste of hex characters into documents. You may edit on either the hex or ASCII side in hex view, and the cursor position is highlighted on the other side to help you track your location.

Project Manager Enhancements

- CF6072, CF5395: The Extra Stack property has been fixed to default to 2K. This had been defaulting to 512K.
- CF4982, CF7418: The order of components is retained when the project file is saved. The components may be reordered through both a mouse and keyboard (<ALT>Arrow) interface.
- CF7417: Units may be reordered to provide greater control of the build process. A new dialog, **Unit Maintenance**, has been created to allow the user to enter new units, delete current units, and reorder all units.
- CF7278, CF6825: The **Exclude from Build** option is fixed.
- CF7082: The **Search** control on the Standard toolbar now works correctly. If you enter text to search and press Enter, the search will take place.

- ICF7229: The Edition # property is retained when the project is saved.
- CF2943: The **Serial** and **Telnet** windows are now stand-alone toolbars rather than tabs on the **Output** toolbar. They may be displayed and docked individually.

Debugger Enhancements

- The speed of the debugger has been increased.
- A new stand-alone debugger (`hawkdbgr.exe`) has been created to handle the debugging of forked child processes. This debugger may be run as a separate process.
- CF694: All debugger data windows, except **Process I/O**, are disabled when the debugger is communicating with the target. The **Process I/O** window is no longer a tab in the Output toolbar. It is a separate window.
- CF7241: The Hawk debugger now looks at the environment variable `TARGETUID` to determine what user ID should be for the process on the target system. The format of a user ID is `ggg.uuu` where `ggg` is the group number and uuu is the user number.
- CF6921: A variable with the volatile type will no longer confuse the debugger symbol browser.
- CF7279: The Target Load and Unlink dialogs now remember up to 8 previous targets used.
- CF4184: The debugger used to have a problem when the user reset the target. The debugger will now gracefully handle the reset of the target.
- CF2946: When the **Watch** or **Locals** windows are resized, the second column will automatically resize to completely fill the window.
- CF6816: The debugger now allows the setting of complex breakpoints. A complex breakpoint usually contains a condition in addition to a location.

Known Problems

- Due to the debugger enhancements, the debugger daemons need to be updated on the target OS-9 machine. The Hawk debugger will sense any incompatibilities and inform the user rather than continue.
- 3007: If the user presses the **Stop** button while the Hawk Debugger is waiting for input, the connection will be lost, Hawk is likely to be temporarily frozen, and the target daemons will be left dangling. This is a limitation of OS-9.

Documentation Notes

The Hawk documentation, in PDF format, includes the following manuals:

- ***Getting Started With Hawk***
- ***Using Hawk***
- ***Using Hawk Macros***

These documents focus on using the Hawk integrated development environment to develop Hawk projects in OS-9—including using the editor, debugger, and compiler.

The Hawk documentation, in standard WinHelp format, includes the On-line Help System, which is accessed from the Hawk interface.

The On-line Help System focuses exclusively on using the Hawk editor, which is a derivative of the Codewright 6.0 editor from Premia Corporation. The On-line Help System may contain references to Codewright functionality that has been removed from Hawk. It may also not include functionality added to Hawk for this release.

Hawk 1.2

For More Information

For complete information on how to use Hawk, refer to the following documentation.

Getting Started With Hawk

Using Hawk

Using Hawk Macros

The Hawk On-line Help System (accessed from the Hawk Integrated Development Environment).

Enhancements

The following issues are believed to have been resolved:

- Hawk better handles variables which have been optimized away to nothing
- General protection fault in compiler front end has been fixed
- Header files will be included in next OS-9/68k product release
- Line number of error is now available in assembler
- Hawk no longer uses an absolute path and does not assume code is on drive C:
- The Error button in the toolbar now works
- “run” and “stop” sometimes entered a mode where they only worked every second time
- Problems with the rebuild button in Project view have been fixed
- Hawk is now able to recover, if the target is rebooted

- Toggling of read-only mode now works correctly
- General protection fault in Linker has been fixed
- Numerous access violations have been fixed
- Hawk no longer has problems moving and docking windows
- The FasTrak manual has been replaced by the Hawk manual
- The access violation in Hawk when using absolute path names is now solved
- Environment variables can now be separated by carriage return
- A general protection fault in `cpfe` is fixed
- The watch window no longer limits an array to just 32 elements
- The `RegDBDeleteKey` error was removed from InstallShield
- Telnet to OS-9/StrongARM now works correctly
- Button/menu problems fixed
- Packaging changed to add definitions files and solve 8.3 file name problems
- A problem where closing one window causes another window to close has been fixed
- Access violation problems when importing project have been solved
- Missing help file is now included
- Bounds checking has been added to array accesses
- A certain key sequence could cause Hawk to enter infinite loop
- The debugger can now find files in sub-directories
- Hawk is now able to attach to subroutine modules
- Hawk no longer has compiler's -t and -s options swapped
- A problem in dependency creation has been fixed
- Mishandling of PowerPC r0 during copy propagation optimization has been fixed
- Left-right scrollbar has been added to “build” window

- Hawk no longer “freezes” after certain combined key presses
- Updates to assembly source code window during animation have been improved
- Problems with default Hawk environment has been fixed

Architecture Notes

The graphical, PC hosted, sections of Hawk have been completely re-designed to reduce the numbers of exception faults and to facilitate easier future maintenance. Other improvements have been made to the host code, in order to improve debugger performance: especially code used for stepping through individual lines of the debugged application.

Documentation Notes

The documentation and on-line help have both undergone significant improvements.

Changes made since the previous product release include:

1. Eliminated Using Hawk Profiler (incorporated into Using Hawk)
2. Eliminated Hawk Programming Reference (now in Help file only)
3. Eliminated Using Hawk Utilities (now in Help file only)

The resulting manual set includes:

1. Getting Started with Hawk
2. Using Hawk
3. Using Hawk Macros

Known Problems

- Target Connection

  In order to connect to the target in user state, you need to run `spfndpd` in the background.

- Installation Location

  Hawk needs to be installed on a computer that does not have a previous version of the Ultra C compiler resident on it.

- Attaching to a Module

  Attaching to a module during a system state debugging session when the debugger connection is not active (host is currently running) invalidates the debugging session. Hawk must be restarted and the target reset before a system state connection can be re-established.

- Moving Projects

  Moving projects into different locations is not available. If a project must be moved, the project's preferences must be changed manually.

- Aborting a system state connection

  Aborting a system state connection is not currently supported in Hawk.

- Setting Break Points

  You may be unable to set break points at some locations, such as the first line of code. To work around this problem, set the break point on main and trace to the first line or you can create a dummy first statement.

- Response Delays

  Hawk may take a few seconds to respond after hitting a break point or the step/next functionality in the debugger when debugging in system state.

- Absolute Paths

  Hawk projects require absolute paths. For example if a project is developed on drive C and moved to drive D, you should modify the paths within the project for all source and output folders of the **Projects Properties** dialog.

- Preloading programs

  Hawk requires programs to be preloaded in the target's memory in order to perform debugging. This implies that if a main process forks several child processes, you must preload all corresponding modules, including the main module in case it forks itself.

- Process forking itself

  Hawk does not allow you to debug a process which forks itself. If you encounter this situation, make a clone of the corresponding module with a different name and modify the main process to call this clone.

- System State and SPF Daemons

  You cannot run system state if SPF daemons are running.

- Attaching Module in System State

  If you attempt to attach a module in system state when the connection is not active (connection is established and **go** has been done), Hawk comes back with an error **cannot attach**. After this, typing break on the target does not restore the system state connection.

- SPF In System State Debugging

  SPF cannot be used over Ethernet during system state debugging. If you start SPF on the target when using Hawk to debug in system state, Hawk will fail to connect to the target. You can use SPF and system state debugger over SLIP instead.

- Output Window

  The Hawk output window does not have a buffer large enough to hold long output. This problem will be addressed in a future release.

- Active sub window

  If any sub window (window other than Hawk main and source windows) is active, the function keys for step/next for assembly/source instructions do not work.

- Stepping

  In the debug window, the function keys to run or start stepping through a program do not work until the cursor is placed inside the window containing code.

- Next

  If the code box is maximized, an icon appears to the left of the **File** menu at the top of the screen. When clicking on that icon, one of the displayed choices is **Next Ctrl+F6**. However, Ctrl+F6 does not work properly.

- Watch Windows

  Global variables of a program do not update in the watch window, but the actual change is performed in the program.

- Lingering Windows

  If you experience lingering windows, such as the properties and compilation windows, do not run Hawk with Perfect Screens. The two software packages do not work together.

- Output Buffer Scroll Bars

  Vertical and horizontal scrollbars in the output buffer return to the original position of the cursor when released. Instead, use the scroll arrows, which work properly.

- Adding/Removing toolbar buttons

  It is possible to remove buttons from the debug toolbar, but you will be unable to add them back to the toolbar.

- Empty debug toolbar

  If the debug toolbar shows up empty with no button icons in it, comment a line containing “Debug” string in `$(MWOS)/DOS/BIN/mwhawk.ini` file and restart the debugging session.

- Buttons not functioning

  The toolbar buttons will not function if the Hawk window background is selected.

- Next error button

  The next error button on the toolbar (looks like three triangles with the letters ERR) clears all errors in the build window at the bottom of the screen and doesn't advance to the next error.

- Question mark icons

  In the Toolbar Customization dialog box, the two question mark icons both perform the same operation, "Help for keyword under cursor."

- Spell Check

  If you select **Tools->Spell Check** you may receive a warning: **Function "DlgSpellCheck" not found**.

- Cleanup Option/Incremental Link

  There is no cleanup option or incremental link option in the current version of Hawk.

- Opening a `.prf` file

  Opening a `.prf` file, followed by starting a profile session in the file's context may cause erratic behavior (like a dead profiler session, module list not displayed, etc.). **Open** is simply to view a previously saved profile. For a new profile, do a **New,** and then **Profile**. If you chose to do an **Open**, it is recommended that you invoke a new profiler session for future profiling.

- Alternate connections

  Alternate connections to the target do not work if the new connection is being established when `spfnppdc` for the existing connection has not exited. This issue is caused by timing related issues on the target.

- Update

  Double/multiple entries are added on clicking the **Update** button several times before the earlier ones are complete.

- Target IP address

  In the absence of a DNS, create a host file called `hosts` in the Windows directory. Sample host file:

  172.16.2.217 <host_name 2>

  172.16.2.45 <host_name>

- Stop to view

  If you select **Stop to view** in the Display Parameter window, you can still view the update information in the main window by pulling the horizontal scrollbar to the far right and then bringing it back to the far left.

- File -> Save As

  The display has an **Open** button instead of a **Save** button. This button does perform a **Save** operation.

- Multiple Sessions

  Multiple profile sessions to a target is not supported.

- Overwrite Warning

  Save does not display an overwrite warning if the file already exists and assumes the working draft can be overwritten.

- Open without file extension

  The file name with extensions has to be specified to open a `.prf` file.

Hawkeye Notes

This section contains release notes for Microware Hawkeye. It includes cumulative notes starting from version 2.0.

Hawkeye 2.1.4

- CF9181: Request HawkEye update to correct large initial file (pipe) size issue. HawkEye had been using an initial pipe size that was almost 500K. Hawkeye has been changed so that the pipe size it uses is now only 3K. Edition numbers of the modules that contained the fix are as follows: `router` #10, `loggerd` #9, `cmdd` #10.

Hawkeye 2.0

Threads Support

HawkEye now treats the thread ID almost exactly the way it used to treat process IDs.

- Timelines are by thread ID
- Interactions are to and from threads

The following new system calls are supported:

`F_THFORK`	Fork a thread.
`F_THEXIT`	Exit a thread.
`F_THREAD`	Other thread services.
`F_WAITID`	Wait for a specific thread or process.

Support for User Event Data Modules Removed

User events in data modules is no longer supported. It is replaced with a set of system calls and library routines that log directly to the main `slm` event log. You can use them as follows:

```
err = hawk_control_log(event, note);
```

`event`	A number that will be reported as the user event number.
`note`	A string up to 32 characters long (including the terminator.)

Support for Time That Wraps Multiple Times in the Log

Note

This feature is not supported between events.

Long logs have been known to cause 16- and 32-bit timer values to wrap. Since the event log is sorted on time, this results in a shuffled event log.

Now the host side of HawkEye computes time across wraps of the clock. It will, however, assume one wrap for any interval when the time decreases. This is wrong, but unavoidable, when the timer actually wrapped multiple times in that interval.

Manual Trigger Added

The "stop capture" dialog box that appears while the HawkEye host software waits for a log to be collected has a new "trigger" button. This sends a "manual" trigger to the target immediately.

Memory System Calls Return Both Address and Size of Allocated/freed Memory

The event data for `SrqMem` and `SrtMem` system calls now return both address and size of memory when appropriate. This is useful for tracking memory problems.

The Module Map Command

The memory map command (button and entry in the tools menu) collects a module map from the target system. It uses the memory map to assign names to processes (the process' program module) and to express PC values as offsets in modules.

Unless the set of modules in the target system is static, information from the module map must be read as hints. The module map is collected at a point in time. The event log is collected over a period of time that does not overlap the module map collection.

Support for Reading HawkEye 1.0 Log Files

HawkEye 2.0 can read HawkEye 1.0 log files. It automatically converts them to 2.0 format when they are loaded.

No Support for Mixed Versions of Target Daemons

HawkEye 2.0 requires HawkEye 2.0 target daemons. HawkEye 2.0 target daemons do not work with HawkEye 1.0.

Changed the Trigger Dialog

The order of immediate and full buffer has been switched (to better model logic analyzer operation).

Moved Focus Off Details

Focus can still be brought to the details box by clicking in it, but it defaults to the event window. This makes it easier to open details and cursor around the events.

Processor Details

Information from the preferences/target page are saved and restored with the event information.

Target Clock AutoSet

The resolution (and range) of the target clock is measured by SLM and returned with the trace information from the target. This overrides any information placed in the preferences/target/target clock field, (but the user can re-override that value after the log has been collected if that seems desirable.)

PersonalJava Solution for OS-9 Notes

This section contains release notes for the Microware PersonalJava Solution for OS-9. It includes cumulative notes from versions 1.1, 3.01, and 3.1.

PersonalJava Solution for OS-9 v3.1 Notes

General Notes

General Implementation Note: PersonalJava Solution for OS-9 v3.1 has been implemented on OS-9 V3.0, which features POSIX-style threads. This means the Java-specific "green-threads" library is no longer used. This enhancement to Microware's PersonalJava Solution improves security and reliability, and provides a less restrictive programming environment for people writing native methods. Standard C library functions can now be called from native methods, and calling such things as _os_sleep() (or any other call that causes blocking) is no longer prohibited.

Enhancements

- CF8202: Error numbers are no longer printed as #000:000. This was caused by not sharing `errno` among the different shared libraries. `errno` is now shared among the different shared libraries.
- CF8233: The `java/net/InetAddress/getLocalHostName()` now returns the correct name. The problem was caused by a bug in the implementation of `getLocalHostName()`.
- CF8328: The installation procedure for PersonalJava Solution for OS-9 on x86 is now the same as for other platforms. x86 users no longer have to manually set environment variables for the PersonalJava environment.

- CF8581: A source of random crashes of PersonalJava Solution for OS-9 on x86 has been eliminated. The crashes were caused by a bug in the x86 assembly-language interpreter. Applications left running for long periods of time, or applications in environments where numerous, frequent interrupts occurred, were especially prone to the problem.
- CF8667: The characters '<', '>', '^', '_', '|', '~', and '@' are no longer missing from the monospaced font and the characters '^' and '~' are no longer missing from the serif and sans-serif fonts.
- CF9560: Applets are no longer denied the use of multiple top-level Frames.
- CF9757: In addition to the MicroType fonts released in previous versions of PJava, we are now also shipping TrueType fonts. TrueType fonts are a higher-definition font; the added detail comes at the cost of additional memory consumption.
- CF9873: PersonalJava Solution for OS-9 now ships with a resource file for 4-bit graphics support (stock_4.res).
- CF10022: PersonalJava Solution for OS-9 no longer crashes when displaying complex graphics and text images using XOR mode. This problem was caused by a failure to release a MAUI draw-mask.
- CF10023: Under certain conditions on specific graphics devices, PersonalJava Solution for OS-9 would hang when rendering text. This was caused by a bug in an underlying graphics layer (AFW), and has been fixed. It no longer hangs in these situations.

PersonalJava Solution for OS-9 v3.0.1 Notes

These features were added to PersonalJava Solution for OS-9 3.0.1:

- Support for all Optional AWT Widgets, including scrollbars and modeless dialogs.
- Support for Profiler and debug interface.
- Ten percent overall performance improvement.
- Updated and enhanced makefiles in port directories.

PersonalJava Solution for OS-9 v1.1 Notes

This section describes the release of PersonalJava Solution for OS-9 1.1. It provides a brief overview of the changes between version 1.0 and 1.1 along with specific changes relating to OS-9. This chapter also describes problems encountered when running PersonalJava Solution for OS-9 1.1.

Enhancements

- Support for remote classes

  PersonalJava Solution for OS-9 1.1 allows classes to be loaded from an HTTP server. See Chapter 10 of ***Using PersonalJava Solution for OS-9 for OS-9*** for details about this feature.

- JavaCodeCompact

  The JavaCodeCompact utility allows Java classes to be ROMed. This feature is included with PersonalJava Solution for OS-9 1.1. See the manual ***Using JavaCodeCompact for OS-9*** for details about using this tool.

- Shared Library Module

  PersonalJava Solution for OS-9 1.1 uses a shared library module to provide graphics/windowing support. This shared library module is named `libmawt.so` (`libmawt_g.so` is the debugging version.) This allows a significant amount of code to be excluded from the system where graphics and windowing is not needed.

Known Problems

PersonalJava Solution for OS-9 1.1 will exit with an `error #103:004` if the system time is set to earlier than 12:00am GMT January 1, 1970 or later than approximately 12:00pm GMT January 19, 2038. If your system does not have a battery backed real-time clock, make sure the system time is set at startup to some value between these two dates.

PersonalJava Solution for OS-9 1.1 will exit with an `error #103:004` if the `CLASSPATH` environment variable is unset or if the core PersonalJava packages (`java.lang`, `java.io`, `java.util`) cannot be found in any directory, `.zip` or `.jar` file pointed to by the `CLASSPATH` environment variable.

SoftStax/LAN Communications Pak Notes

SoftStax v3.5

All networking components are now listed under SoftStax. So this section contains release notes for both SoftStax 3.5 and the LAN Communications Pak 3.5.

Softstax 2.2.2 and LAN Communications Pak 3.3 were re-versioned as SoftStax 3.5 and LAN Communication PAK 3.5. This was done to simplify version tracking.

Libraries

- All network libraries are now thread safe.
- `mbuflib.l`

  CF8355: A problem in the mbuf library that could corrupt mbufs with the `SPF_NOFREE` bit set was fixed.
- `socket.l`

  The `recv` and `recvfrom` socket library functions now support the flags parameter. The supported flags are `MSG_PEEK` and `MSG_WAITALL`. These options also require the new protocol modules to work correctly.

  The socket library functions `getsockname()` and `getpeername()` now return the correct information after doing a non-blocking accept. This fix also requires new protocol modules.

  CF7738: The `setsockopt()` function in the socket library now accepts either an `int` or a `u_char` as a parameter for the `IP_MULTICAST_TTL` and `IP_MULTICAST_LOOP` socket options. This fix also requires new protocol modules.

Protocol Modules

- Mbuf usage has been improved and socket programs will now consume less mbuf space per active connection.
- CF4888: An ICMP error in response to a TCP SYN packet could potentially cause a continuous stream of SYN packets being retransmitted. This problem has been fixed.
- CF6718 and CF2734: Much of `spenet` was rewritten to improve performance. This also fixed known problems with mbuf leaks and arp flooding.
- CF9668: A problem with non-blocking sockets was fixed. In some situations a byte count of 0 was being returned, when in fact some of the data written was actually sent. The returned count of the number of bytes written is now correct in all situations.
- CF10066 and CF10448: Resetting TCP connections could cause SPIP to panic and crash the system in certain low mbuf situations. This situation could also result in an mbuf leak. Both problems have been fixed.
- Support has been added to allow SPIP to close and remove interfaces. This is accomplished using the new start, stop, unbind, and rebind options for `ifconfig`.
- CF9583: The zero copy TCP support caused PPP to leak mbufs when sending TCP data. This leak has been fixed.
- Assigning of local and/or remote IP address for PPP causes undesirable results including system resets. Workaround: Obtain IP address from the peer on the remote end of the PPP link.
- The PPP stack has been replaced with a more complete and efficient stack.
- A `ppplib.l` library has been provided that may be linked from within the user application to configure the stack.
- The pppd utility may be used to set up, configure, and tear down the stack as before.

- Using existing chat script files will not work. Change them based on the chat script commands section. New chat script commands have been added.

Utilities

- bootpd

  CF8645: The bootp server now works correctly with multiple Ethernet interfaces.

- ftp

  CF9603: FTP no longer requires the use of the `-s` option when specifying `/nil` as the destination for a get command.

- ftpd

  The FTP server, `ftpd`, now supports `-f` and `-e` options to specify alternate directory listing commands. Also, directory listings are always sent in ASCII mode even if the current mode is binary.

- ifconfig

  Four additional commands were added to `ifconfig` to support the removal of interfaces. The commands are `stop`, `start`, `unbind`, and `rebind`. The `stop` command causes the hardware device associated with the specified interface to be closed. The interface remains in SPIP's internal interface table and may be restarted using the `start` command. The `unbind` command is the same as `stop` except it also causes all information about the interface to be removed from SPIP's interface table. The `rebind` command can be used with a stopped interface to associate it with a different hardware driver stack.

- ping

  CF9138: Ping no longer exits immediately when receiving ICMP packets other than the expected response.

- rpcdump (Windows hosted)

  CF8840: The Windows hosted version of `rpcdump` now correctly prints the contents of an `rpcdb` module.

- rpcgen (Windows hosted)

  CF8509 and CF8655: There is now a Windows hosted version of `rpcgen` in addition to the resident version.
- tftpd

  The tftp server (`tftpd`) now supports the block and file size options specified in RFCs 2347, 2348, and 2349.

Ethernet Drivers

- `sp82596`

  CF8637: The `sp82596` driver has been fixed so it no longer hangs or crashes the system under heavy packet load.
- CF8532: The MTU for the QUADS port is now correctly set to 1500 instead of 1496.
- CF7725 and CF2733: The `spquicc` driver has had several changes. An mbuf leak when transmitting packets was fixed, multicast support was added, and several other minor bug fixes. In addition it has been changed to make it easier to port to other boards.
- CF10042: The `sp21140` driver now supports Netgear cards.
- CF8260: Multicast support was added to the `sp21140` driver.
- CF8554: The `spquicc` driver did not work correctly with multicast addresses where the 4th byte is greater than 127.
- CF9967: The `spquicc` driver now checks `lu_txoffset` and allocates a new mbuf if there is not enough space. Also, cache invalidates were changed to reduce unnecessary invalidates.

NFS

- CF7894, CF8602, CF8855, and CF8902: The 2GB file system limit for NFS has been removed. Any RBF file system may now be exported regardless of size.

- CF8904: mountd now prints an error instead of crashing with a bus error if it is unable to resolve the name returned from gethostname() into an IP address.
- CF3062: The NFS server no longer maintains exclusive access to a file for 30 seconds after a write. The file can now be accessed (read only) on the host immediately after a write operation by nfsd.

SoftStax/LAN Communications Pak (previous versions)

This section contains release notes for the Microware SoftStax and LAN Communications Pak networking software. It includes cumulative notes from versions 3.3 and 3.3.1 of the LAN Communications Pak and cumulative notes from version 2.2.1 and 2.2.2 of SoftStax.

Note

SoftStax was formerly titled SPF Base Package. References to SPF Base Package should be understood as SoftStax.

Libraries

- socket.l

  The getsockname and getpeername functions were changed to do a getstat to spip instead of relying on the ITEM data structures to be kept current.

  The send function now uses a setstat to the SPF file manager instead of being converted into an _os_write() call. This is similar to the way sendto has always worked. The result is a slight speed improvement but also provides the capability to pass the flags parameter. Currently, the flags are still not used however.

The `sendto` call now performs an `_os_chkmem` call on the buffer being passed. This prevents invalid pointers from being passed which could crash the system as well as closing a potential security hole.

The backlog parameter to listen was ignored and assumed to be the old maximum of 5. It is now passed correctly, and the maximum has been changed from 5 to 128.

The `accept` function properly sets the foreign port and address so subsequent `getpeername` functions work correctly.

- `netdb.l`

  Fixed problem with the trap module not handling the stack properly in certain situations.

  If a DNS server returns an ICMP port unreachable error, the next listed DNS server is tried rather than returning an error.

- `rpc.l`

  Now compiled with long code and long data for all targets.

Protocol Modules

- `spip`

  Get IP address from low level if booted with `bootp`
  `getsockname/getpeername` now here instead of in file manager

  Multicast support has been added.

  Support for IGMP version 2 has been added.

  Now able to add interfaces and routes from an `inetdb` module in ROM.

  Correctly counts input bytes on the loopback interface. This value is now correct when viewed with `netstat`.

  The `checksum` function is now written in assembly for ARM4 targets. This is the common function also used by SPTCP, SPUDP, and SPRAW.

If only an interface's netmask changes and not its address, the request is not passed to `spenet`.

Applied patch for CERT advisory CA-98.13.

The 500ms IP timer now only runs when IP fragments are present in the IP reassembly queue.

- `sptcp`

  The `mbuf` to `mbuf` data copy in `sptcp` has been removed. Now the only time the data is copied is when it is copied from the users buffer into an `mbuf`. This results in a substantial improvement in TCP performance.

  On PPC platforms, it also does a `checksum` during the copy into an `mbuf` providing another increase in performance.

  The retransmit timer is set if TCP can not get an mbuf to prevent certain deadlock situations when running low on available mbuf space.

  If all TCP sockets are closed its 500ms and 200ms cyclic timers are stopped. This allows the system to suspend if power management is enabled.

  The TCP congestion window now opens correctly, rather than too quickly as before.

- `spudp`

  Support for multicast socket options added.

  Incoming data is now delivered correctly to multiple recipients, even if it is in an mbuf packet chain.

- `spraw`

  No longer corrupts memory when binding a RAW socket to a specific IP address.

  Incoming data is now delivered correctly to multiple recipients even if it is in an mbuf packet chain.

  Connected RAW sockets now work.

- `sproute`

  Verifies size of user-supplied buffer before copying interface information into it.

- `spslip`

  Added support for IP multicasting.

  No longer corrupts occasional non-TCP packets by attempting to compress them.

- `sphdlc`/`splcp`/`spipcp`

  Now allows multiple PPP stacks to coexist by changing the `lun` number in the descriptors.

- `spipcp`

  Added multicast support.

Descriptors

Using OS9make `make clean` now correctly causes a descriptor to be rebuilt. It is no longer required to do a `purge` in order to force a rebuild.

Utilities

- arp

  Uses `spenet`'s new `arptab` structure, will no longer work with previous versions of `spenet`.

- dhcp

  Now uses the `SERVIDID` option in the `DHCPOFFER` packet if possible. This makes `dhcp` now work correctly with a Windows NT DHCP server.

  Fixed potential bus error when adding DNS servers if the number returned from the DNS server does not match the available space in an `inetdb` module.

- ftpd/ftpdc

  Now sets the `SO_REUSEADDR` socket option on the listening socket. This allows the server to be restarted even if some sockets are in the `TIME_WAIT` state.

  In some error situations, the server would send duplicate error messages and get into a confused state. These duplicates have been removed.

- idbdump

  Now prints `mw_flags` in the interfaces section. The override flags are: `iff_nopointopoint`, `iff_nobroadcast`, and `iff_nomulticast`.

- idbgen

  Modules are now created as a multiple of 8 bytes in size to fix alignment problems on some platforms.

  Added `iff_nobroadcast`, `iff_nomulticast`, and `iff_nopointopoint` options to override driver default values.

- ifconfig

  Added `iff_nobroadcast`, `iff_nomulticast`, and `iff_nopointopoint` options to override driver default values.

  Repeated changing of an interface's netmask no longer results in an `EOS_FULL` error.

- ndbmod

  Added `iff_nobroadcast`, `iff_nomulticast`, and `iff_nopointopoint` options to override driver default values.

- netstat

  Added `-ia` option to print all multicast groups joined on each interface

  Prints `igmp` statistics using either the `-s` or `-p igmp` options.

- ping

  Now exits immediately instead of waiting when `sendto()` returns an error such as `EHOSTUNREACH`.

  Increased the size of the receive buffer to handle packets bigger than 48K. Now able to specify values up to 65507 using the `-s` option.

- pppd

  Fixed hole where we can receive a signal before we are ready to process it.

- rpcdbgen

  Modules are now created as a multiple of 8 bytes in size to fix alignment problems on some platforms.

  exportfs / showmount / rusers

  The date printing is now year 2000 compliant.

- rusersd

  Now uses 1970 as its time epoch instead of the OS9 version 1.x time epoch of 1980.

Ethernet Drivers

- `spenet`

  If `spenet` does not have enough space to contain the Ethernet header, it allocates a new `mbuf` and copies the data into it. Previously it did not leave any free space for additional header required by the hardware drivers. Now it leaves the amount of free space requested by the hardware drivers `TXOFFSET` variable.

  Now supports IP multicast packets.

  Correctly sets `arp_op` to `ARP_REPLY` when responding to proxy `arp` requests.

  Fixed mbuf leak when receiving packets from broken hardware drivers.

SPF File Manager

- The receive thread now inherits its priority from the starting process rather than being fixed to 128.
- Added `FMCALLUP_TIMER_START`, `FMCALLUP_TIMER_STOP`, and `FMCALLUP_TIMER_RESTART`. These macros take the same parameters as the following functions and are intended to replace them: `timer_start`, `timer_stop`, and `timer_restart`. These functions are found in `sptimer.l`. For compatibility, the library is still available and now uses the file manager callups.
- Fixed a problem where the SPF timers could become very inconsistent when using a combination of one shot and cyclic timers.

General Improvements

- sysmbuf: The module name has been changed from `SysMbuf` to `sysmbuf`.
- CF2602: The FTP server (`ftpd/ftpdc`) now supports a `-f` and `-e` option which can be used to allow a web browser to correctly view an OS-9 target.
- CF2731: A memory leak in `spenet`'s `dr_term` has been fixed.
- CF2734: `spenet` has been changed to prevent `arp` flooding. At most 1 `arp` request for a particular address will be sent per second. Also, the length and resolution of the timer that controls removing `arp` entries is configurable in the `enet` descriptor. The default values are 60 seconds for the timer interval, 20 intervals (i.e. 20 minutes) to remove completed entries and 3 intervals to remove incomplete entries.
- CF2769: `Ifconfig` no longer requires UDP. If it can not open a UDP socket it will try a TCP socket, and if that fails it will try a raw socket.
- CF2784: `sptcp` was incorrectly adding an additional maxseg / 8 when opening the congestion window. It now uses the correct value of (maxseg ^ 2) / cwnd.

- CF2789: `ipstart` now prints a warning instead of an error if no protocol modules (`sptcp`, `spudp`, `spraw`, or `sproute`) are successfully started. This is most often cause by a missing descriptor so it also reminds the user to check and make sure they have the correct drivers and descriptors loaded.
- CF2801: The daemon `routed` will no longer crash if an interface is added via `ifconfig`.
- CF2853: The SoftStax porting guide contained some typos in the section that discusses hold-on-close. This section is in the **Driver Considerations** section of the **Creating SoftStax** chapter.
- CF2920: This is a duplicate of CF2853.
- CF2891: Fixed a problem with select and TCP sockets. There was a small window of time where a socket being closed by the remote end would not break an application out of a select call. This hole has been fixed and now select should return in all situations.
- CF3056: The "output packets discarded due to no route" counter returned by netstat does not get incremented. This is not true, it is incremented if the packet originated on the host. If the host is acting as a router it should return an `ICMP_UNREACH` error to the send, but not increment the counter. This is the current behavior.
- CF4842: The ftp `mget` command fails to work if the current directory is on a PCF device. This was caused because the temporary file created to hold the file list did not conform to the 8.3 naming standard. This has been changed to a shorter filename and now `mget` works correctly.
- CF4888: ICMP Destination Unreachable message can cause packet storms. An incoming `ICMP_UNREACH` message in response to a SYN packet caused the SYN to be incorrectly retransmitted. This in turn generated another ICMP error and a resulting packet storm. This behavior has been fixed.
- CF5111: The check in `sptcp` to retract the congestion window was incorrect. The old check was: `tp->t_dupacks > tcprexmitthresh` but has now been changed to the correct: `tp->t_dupacks >= tcprexmitthresh`.

- CF5337: The mkdir command in the FTP server (ftpdc) returns an EOS_PARAM error. This has been fixed and now the mkdir command works correctly.
- CF5386: The TCP/IP stack runs out of mbufs and stops working under high load. It is not an error that in a situation where you have a mismatch between the size of the mbuf pool and the amount of traffic that all TCP traffic will pause. However, this pause should only last until TCP connections begin timing out. In some situations connections timing out would not free their allocated mbufs and the pause turned into a deadlock situation. This is now fixed and any terminated connection will return its mbufs and the system will continue to run normally.
- CF5605: The header files auth.h, auth_unix.h, os9.h, pmap_clnt.h, pmap_prot.h, pmap_rmt..h, rpc.h, rpc_msg.h, svc.h, svc_auth.h, types.h and xdr.h in the MWOS/SRC/DEFS/SPF/RPC directory have been modified to work correctly when compiling with C++ code.
- CF6564: The java function getLocalAddress() and socket function getsockname() do not return the correct value. When doing a non blocking connect, the value returned from a subsequent call to either of these functions returns an incorrect value. This has been fixed, and they now return the correct value in all situations.
- CF6718: In some error situations spenet will leak mbufs. These have now been fixed.
- CF6785: Simultaneous FTP commands result in an EADDRINUSE error. This is now fixed, and both commands will complete normally.
- CF7466: Parsing problem because of extra semi-colon in sys/cdefs.h when using C89 or ANSI modes. The stray semi-colon has been removed from the cdefs.h header definition of __END_DECLS.
- Added capability for select() call to return if data is already waiting on a path.
- The SPF file manager chains application data for datagram I/O option path (spf_popts.pd_iopacket type is IO_PACKET_TRUNC), and sends it down to the protocol driver.

- The SPF file manager checks the TCP read status (`pd->pd_item.error_state`) before going to sleep to wait for data within `read()` function, and returns if there is an error state.
- `item.h`

  Added `ITE_ON_DR_DEFINE` for `notify_type.ntfy_on` for driver specific notification type.

  Added ethernet address class `ITE_ADCL_ENET` on `addr_type.addr_class`.
- `item_pvt.h`

  Added `ITE_GETADDR` definition.
- `mbuf.h`

  Added `MBT_SENDTO` definition for `mbuf.m_type` to distinguish UDP `sendto()` and `send()`.

  Added `M_PREPEND` macro to prepend space size requested to mbuf if there is enough space.
- `spf.h`

  Added `SPF_PR_SOCKET` definition to the Microware protocol type.

  Added `spf_pdstat.pd_flag` to support UDP `sendto()`.

  `pd_ioasync` path option bug is fixed: `IO_SYNC` is 0, `IO_ASYNC` is 1, `IO_WRITE_ASYNC` is 3, and `IO_READ_ASYNC` is 5.

  Added `SPF_PR_USB` definition.

  `lu_prottype` in `spf_luopts` structure (Logical Unit Option) is moved to after `lu_optsize` to accommodate `u_int16` definition.
- `stat.c`

  `lu_prottype` field in `spf_luopts` was moved to after `lu_optsize`, and `lu_rsv2` field (renamed from original `lu_prottype`) has 0 value. All descriptors in the SPF Base Pak v2.2 were remade with the new `stat.c` and are shipped in SoftStax Base Pak v2.2.1.

- desc.tpl

  os9make clean does nothing, and os9make purge removes the target descriptor. It is necessary to purge the target descriptor in order to remake it.

- dbg_mod.l

  The dbg_mod.l for ARMV3 didn't line up the debug module static area correctly. This is fixed by the compiler and the pointer to the static area is .rr7+64.

- item.l

  Separated functions out of single library source to provide optimal libraries. Previously, all functions were in a common C file and would be embedded in the final application. By separating each function into its own file, only required functions would be placed in the final application. Added check for NULL pointer in the following functions.

```
error_code ite_ctl_connect(path_id path, Addr_type our_num, Addr_type
    their_num, Notify_type npb);
error_code ite_ctl_connstat(path_id path, Device_type dev_info);
error_code ite_ctl_resadd(path_id path, Notify_type npb);
error_code ite_ctl_rcvrasgn(path_id path, Addr_type telno,
    Notify_type npb);
error_code ite_dev_getmode(path_id path, u_int16 *mode);
error_code ite_ibreschg_asgn(path_id path, Notify_type npb);
```

  The following ite_mpeg functions were removed from the item library:

```
error_code ite_pgm_psiget(path_id path, Ite_psi_pb psi_pb);
error_code ite_pgm_psirmv(path_id path, Ite_psi_pb psi_pb);
error_code ite_pgm_psimask(path_id path, Ite_psi_pb psi_pb);
error_code ite_pgm_notify_asgn(path_id path, Ite_psi_pb psi_pb,
   Notify_type npb);
error_code ite_pgm_send_notify(path_id path, Ite_psi_pb psi_pb);
error_code ite_pgm_notify_rmv(path_id path, Ite_psi_pb psi_pb);
error_code ite_pgm_get_pref(path_id path, Ite_stream_pref pref);
error_code ite_pgm_set_pref(path_id path, Ite_stream_pref pref);
error_code ite_pgm_info(path_id path, Ite_pgm_tbl pgm_table,
    Notify_type npb);
error_code ite_pgm_info_rmv(path_id path, Ite_pgm_tbl pgm_table);
error_code ite_pgm_set(path_id path, Ite_pgm_tbl pgm_table,
    Notify_type npb);
error_code ite_pgm_abort(path_id path);
error_code ite_flush_pat(path_id path);
error_code ite_set_mbanchor(path_id path, Mbuf *mb_anchor);
```

- mbuflib.l

  When the SPF_NOFREE bit is set, the following functions will reinitialize the mbuf header.

  ```
  Mbuf  m_free(Mbuf mb);
  int32 m_free_m(Mbuf mb);
  Mbuf  m_free_p(Mbuf mb);
  void  m_free_q(Mbuf *queue);
  ```

- mbinstall

  Fixed to restore stdin, stdout and stderr.

- mbdump

  Fixed to restore stdin, stdout and stderr.

- SysMbuf

  Removed user state accesses. SysMbuf call for mbdump now returns pool address instead of bit map address.

  Access/Creation is limited to a system state process. Speed optimization for mbuf allocation is added to PPC and ARM.

- mbdump

  Updated help text and formatted bit map display.

- Loopback driver (sploop)

  If a path registers a notification for far-end hang-up, and the connected path hangs up, the far-end hang-up notification registered path now handles the path close properly. A path which has an established connection returns EOS_DEVBSY for ite_ctl_answer call.

- Compiler Optimization Issue for System State Process

  Any spf drivers using item.l, ppstat.l and sptimer.l need to recompile with -ao=sc optimization option due to an icode linking issue in system state modules.

- makefiles

  Cosmetic changes were made in makefiles for WINDOWS support.

- Compiler Warning Issue

  Cosmetic changes were made to most of the networking-related source files to remove compiler warnings. This may have caused the CRC to change.

Enhancements

The LAN Communications Pak is now year 2000 compliant.

Multicast support has been added to the core protocol stacks, as well as the slip and PPP drivers. Refer to the release notes for a particular board for information on multicast support for specific hardware drivers.

Interface drivers now inform IP of their capabilities, such as multicast or broadcast support, rather than depending on their capabilities being specified correctly in the `inetdb` modules or on the `ifconfig` command line. If it is not desirable to use all the capabilities of a driver, new options have been added to `ifconfig`, `ndbmod`, and `idbgen`. For example, if you do not wish an ethernet interface to support broadcasting, you can specify the `iff_nobroadcast` option.

- The SPF file manager checks the persistent incrementing events using `notify_type->ntfy_class` instead of `notify_type->ntfy_on` in `DR_FMCALLUP_NTFY`.
- The SPF file manager correctly updates the read queue count.
- `Dr_use_cnt` has been corrected to count exact number of use count.
- The stack cleanup procedure removes mbufs in the SPF receive thread queue when the network stack is torn down.
- In the case of the stack initialization error at a path open time, the file manager now call cleanup for the bottom driver (normally the hardware driver), and the hardware driver should be removed from the OS-9 system. However, because of the nature of the DPIO, the hardware drivers' attach count will be one more than actual count, and the hardware drivers `dt_term` won't be called for the OS-9 for 68K system. For the OS-9 for 68K system, iniz the devices before running applications is recommended.

The SPF file manager will return `SUCCESS` when partial data has been sent out and there is an `EWOULDBLOCK` error occurred on an ASYNC mode (non-blocking) path.

Known Problems

- Installation

  Care should be taken when installing the LAN Communications Pak on a 68000, PPC, or 80x86 platform which previously had ISP installed. ISP (Internet Support Pak) is the BSD v4.2+ TCP/IP stack provided in the past.

  The two networking packages can not be run on the same machine at the same time, nor can the utilities for one run with the other. In addition, several utilities such as `telnet` and `ftp` share the same names. The installation program will list the conflicting utility names. All conflicting utilities are in `MWOS/<OS>/<PROCESSOR>/CMDS`. The ISP versions should either be renamed or moved to another directory.

  Running any utility with the `-?` option will tell you what networking stack it is for. The LAN Communications Pak utilities will say SPF for the IP stack. If the utility does not list any IP stack, then it is an older ISP utility.

- NFS client and server

  The NFS server only supports RBF devices. PCF file systems can not be exported at this time.

  The `rename()` function returns `000:208` Unknown Service Routine when used on an NFS disk. The workaround is to use the `_ss_rename()` function. Append mode is not supported.

  Mount disk points are limited to 2 Gigabyte media size.

  NFS does not support file locking.

  The `create()` function acts like `creat()` for the NFS server software.

1394 SDK for OS-9 Release Notes

The following provides general issues concerning IEEE 1394 SDK version 1.0.

Transaction Layer and Bus Management Issues

The following items detail issues concerning the transaction layer and bus management:

- Desired IEEE 1212 offset must be provided when requesting address mapping by the 1394 stack.
- Transmission of Link On and extended PHY Configuration packets are not supported.
- Applications are informed about response timeouts for initiated requests. No re-transmissions are performed. It is up to the application to re-transmit, if desired. Responses that are not `ack-ed` successfully are not re-transmitted.
- Block transfers that are not multiples of four are not supported.
- No address validity check is made for 64-bit lock operations; it is the user's responsibility to ensure that a 64-bit address region is mapped when performing a 64-bit lock operation. The 1394 stack currently checks only for a 32-bit mapped region. Attempting a 64-bit operation, having mapped only a 32-bit address range, may cause undesired results.
- `initiated_reset` and `contender` fields of `link_info` structure are not currently updated.
- Currently, the stack does not provide any means of gap count optimization other than that requested at the time of an attempted root.

Link Layer Driver issues (sp8412)

- Since the hardware does not provide the config ROM information in actual ROMs, the BUS_INFO_BLOCK of the config ROM is implemented in the device descriptor. To modify it, change the value of the SPF_LUSTAT entry myConfigROM structure, within SPF_LUSTAT_INIT in defs.h. Then rebuild the descriptor.
- After a bus reset, the SH hardware is unable to respond immediately to lock request sent by a node. Thus if the SH target is being used as an IRM, and a bus manager candidate attempts to lock the BUS_MANAGER_ID register, the lock operation fails. According to Hitachi feedback, a delay of 120ms before initiating this lock request should fix the problem.

Link Layer Driver issues (sp8413)

- The sp8413 driver does not get a lock response because the chip finds a DATA/CRC error in the response packet and returns an ACK_DATA_ERROR and drops the packet. However, this happens only when the lock request did not return with response complete. This appears to be a problem in the hardware.

Using IEEE1394 SDK with the OS-9 Configuration Wizard

You can use the OS-9 Configuration Wizard to build boots that include modules from the IEEE 1394 SDK. To enable this, it is necessary to complete the following steps:

Step 1. Open the Configuration Wizard in Advanced mode.

Step 2. Open the PORT menu from the Sources menu and select User [user.ml] MASTER]. This allows you edit the contents of the user.ml file. Entries added to this file are merged into the boot by the Wizard.

Step 3. Add the IEEE 1394 modules you are using to the end of `user.ml`. The following is recommended as a starting point for the Hitachi MS1394DB0 board:

```
../../../CMDS/BOOTOBJS/SPF/sp8412
 ../../../CMDS/BOOTOBJS/SPF/spff0
../../../../../CMDS/BOOTOBJS/SPF/sp1394
../../../../../CMDS/BOOTOBJS/SPF/sptr0
../../../../../CMDS/1394info
```

Step 4. Close and save the `user.ml` file.

Step 5. Click the `System Network Configuration` button and select the `SoftStax Setup tab`. Select `Enable SoftStax`.

Step 6. Click `OK`.

Step 7. Click the `BL` button and select `User Modules` so the Wizard includes entries in the `user.ml` file. Select `SoftStax (SPF) Support` as well.

You may now use the `Build` button to create a boot image as you normally would.

USB Peripheral SDK for OS-9 Release Notes

General Issues

The following demonstrates general issues to consider when using version 1.0 of the USB Peripheral SDK:

- Per-path storage mechanism is not yet used. In addition, all drivers are designed to use 3 endpoints: the control endpoint, a Bulk IN endpoint, and a Bulk OUT endpoint.

The following issue concerns use of `spusbd823`:

- When using Hawkview, sending files from the Windows machine to the USB device does not work because of a hardware problem. The driver has only been tested on a MPC850 (1F98S) microprocessor.

The following issues concern use of `spusbdsa`:

- SA-1100 revision G or later is required.
- If data is not ready in enough time, SA-1100 stops answering IN requests after several frames. A software work-around has been implemented using a vendor request.
- When run over 192 Mhz SA-1100, USB hardware appears to become unstable. The SA-1100 should always be run at 192 Mhz when USB is being used.
- SA-1100 errata requires that the USB cable be connected when accessing the `SAUDC` register set. The USB cable should be connected to the host and the device when the driver is initialized.

MAUI Notes

This section contains release notes for MAUI, the Microware multimedia API. It includes cumulative notes from versions 2.2, 2.3 and 2.4.

MAUI v3.1.1 Notes

Enhancements

- Re-implemented the CDB and MSG APIs to execute in system state via an MFM driver (`mauidrvr`). There are new chapters in the MAUI Porting Guide that describe these new objects.
- Moving the CDB to system state resulted in a huge speed improvement for the most common CDB call `cdb_get_ddr()`.
- The CDB API no longer limits the maximum size module directory it can read.
- Added `cdb_get_size()` and `cdb_get_ncopy()`.
- Under OS-9 platforms other than 68K, the CDB API now searches multiple module directories.
- The MSG API is now more secure, in that the `mauidrvr` can clean up if an application exits without properly closing its mailboxes first.
- Added `msg_send_watch()` and `msg_release_watch()` to allow MAUI applications to request a signal when another MAUI application terminates.
- Removed support for `msg_set_filter()` in the default version of `mauidrvr`. This function can be used by utilizing alternative versions of `mauidrvr` that are included in the release, but they are not as small or fast. The three drivers (including the default) are as follows:
  `mauidrvr` - default,
  `mauidrvr_lock` - supports record locking,
  `mauidrvr_filter` - supports record locking and msg_set_filter().

This is discussed in detail in the MAUI Porting Guide and the msg_set_filter() page of the MAUI Programming Reference manual. Also see the Using MAUI manual.

- Moving the MSG API to system state corrected a potential erase condition between multiple "simultaneous" mailbox opens and creates.
- Added `gfx_get_dev_capexten()`, `_os_gs_gfx_devcapexten()`, `GFX_DEV_CAPEXTEN_VALIDATE()`, `GFX_VPC, GFX_VPDMC`, `GFX_DEV_MODES`, and `GFX_DEV_CAPEXTEN`. These new functions, macro, and data structures provide additional graphic device capabilities information.

Resolved Problems

- Corrected filter callout bug (introduced in MAUI v3.0) in `msg_flush()` to set and restore user global and constant pointers when using the shared library.
- It was possible to specify a draw clip area outside of a drawmap resulting in the draw context's internal [x|y]min draw bound exceeding the [x|y]max draw bound. This led to occasional address violations. This case is now detected when recalculating the clipping region and used to mark the draw context as fully clipped. This problem was identified by a poorly written JAVA based web browser.
- Fixed a bug in `maui_win` that caused duplicate key/pointer messages to be sent to the application if the application requested a message signal and the signal queue of the application was full.

MAUI v3.0 Notes

Enhancements

- The biggest new feature to MAUI v3.0 is thread support. Extensive changes were made to minimize thread local static data and to move constant data into code space. Unfortunately, some thread local static data is required, resulting in some speed penalties.
- There are now two versions of the MAUI shared library modules. Their file names are `maui` and `mt_maui`. Their internal module names are both `maui`. See the MAUI Porting Guide for more information. Old pre-thread MAUI applications that were compiled to use the MAUI shared library should be compatible with either shared library.
- As part of enhancing the OS-9 kernel to support threads, semaphores now must be explicitly initialized. In the past, you could get away with just defining a semaphore, clearing the memory, and make `sema_p()` and `sema_v()` calls. MAUI is a heavy user of semaphores, but it did not always call `sema_init()` and `sema_term()`. This means that most statically linked pre-thread MAUI applications will not work with a threaded kernel. To correct this, simply recompile your application using the current version of MAUI. As always, it is suggested that you link against the MAUI shared library instead of statically linking to MAUI to avoid this in the future.
- In most cases, MAUI does not have thread specific data. Developers should assume that MAUI globals are application wide.
- MAUI init and term calls are not thread specific. They apply to the whole process.
- BLT, DRW, TXT, etc. contexts should be created, used, and destroyed in the same thread. If contexts are passed between threads, the application is responsible for appropriate locking of calls.
- WIN APIs must lock and unlock regions, and draw in the same thread that created the TXT and DRW contexts.

- Error messages were modified to include both _mainid and _procid. For instance: MAUI Fatal(8|9): 10:36 detected in win_create_dev. indicates that thread 9 of process 8 caused an error.
- New demo_init() and demo_term() functions added to mauidemo.l.
- Enhanced 68k version of CDB API to work with ssm.
- Added read and write entry points to mfm, the sound and graphic driver common code, and to the sample driver code.
- Added new input key codes.
- Reduced the size and time required to create a BLT context by not calculating expansion tables unless they are needed to do an expansion blit. This saved almost 2K per context as well as the time required to initialize the tables.
- Created maui_vfprintf() and maui_fflush() callbacks from the shared library. This allows the shared library to print to stderr without pulling in the fprintf() code. Functions such as mem_list_segments() and mem_list_overflows() now work as documented in the shared library. Developers that don't wish the shared library to print or don't wish to incur the code size overhead of fprintf() functions in their applications can simply define the following functions:

```
#include <stdio.h>
int   maui_vfprintf(FILE *fp, const char *fmt, va_list ap)
{
   return 0;
}

int   maui_fflush(FILE *fp)
{
    return 0;
}
```

- Added SDV_IRQ_LEVEL to audio drivers to differentiate between the IRQ number and level. Previously they used SDV_IRQ_NUM for both. This is a change to sdv_abort.c, sdv_cont.c, sdv_pause.c, and defs.h in the SD_COMM directory. For compatibility with the old drivers, SDV_IRQ_LEVEL is assigned to SDV_IRQ_NUM in case it has not been defined in mfm_desc.h. Use of SDV_IRQ_LEVEL is a required change for sound drivers on SH processors.

Resolved Problems

- mem_list_overflow() now correctly returns the error code EOS_MAUI_DAMAGE if a memory overflow is detected.
- Rewrote all *_init() and *_error() calls to use less code.
- If win_init() gets an error, it now correctly calls gfx_term() and drw_term().
- Corrected a problem in cdb_get_copy() that caused a bus error if the MAUI shared library was not installed.
- Fixed a drw_arc() bug that caused a bus trap when drawing an arc off the bottom of the screen.
- Changed the error type from MAUI_ERR_NON_FATAL to MAUI_ERR_WARNING for cdb_get_ddr() when it returns EOS_MAUI_NOTFOUND. This allows applications to filter out this common and normal error message.
- Changed shared library bindings for ARM to support larger applications.
- Several demos and the winmgr incorrectly used _os_rte() with signal handlers not installed with _os_intercept().
- The comments and macros for snd_set_cm_bit_order(), snd_get_cm_bit_order(), snd_set_cm_byte_order(), and snd_get_cm_byte_order() did not match in maui_snd.h. In theory, the comments were right and the code wrong, but to preserve compatibility with prior releases, the comments are now set to match prior shipped code.

- Corrected names of functions reported by CDB API errors.
- Modified auplay.c and aurecord.c demos to try to find the audio device name from the CDB.
- Changed the CDB APIs default debug level so MAUI_ERR_WARNINGS are not printed.
- Corrected blitting problems associated with blt_expand_block() and blt_expd_next_block() when using expand mode BLT_MIX_RWT on little endian processors.
- IO_BLT drivers which do not support expand using mix mode RWT did not test for RWT and returned an error. Updated gdv_blt.c to detect the unsupported expand using RWT mode and return EOS_MAUI_NOHWSUPPORT. If an OEM customer wishes to support expand using RWT in their driver, they can remove the test in the set_dispatch() function found in gdv_blt.c, and add the required support in gdv_expd.c. This is not in the standard IO_BLT driver code in the interest of size and speed.
- Corrected rasterization problems with drw_arc() and drw_circle(). It was incrementing the center points and not decrementing the radius.
- Fixed an address violation in _gdv_set_topdev() of GX_COMM/gdv_dev.c by protecting the call with a semaphore in open and close. The system state open/close calls were interfering with the user state update display call. This problem was noticed when "fdraw&fdraw" caused an 103:004 error on a ThinClient.
- Corrected GX_VGA to use gfx_get_cm_name().

MAUI v2.4 Notes

The following features were added to MAUI 2.4:

- New 1 to 2 bit expansion in driver IO_BLT.
- New True Color swap bit in GFX_CM allows better support of 16 and 32 bit color hardware.

- New `gfx_*_dev_attribute()` calls allows setting of brightness, contrast, hue, saturation, sharpness, gama, white balance, etc. on some hardware.
- Improvements in oval and arc drawing.
- Corrected copy block problem that could result in data access outside of a drawmap.

MAUI v2.3 Notes

This section contains release note information for MAUI 2.3.

Note

Because of the limited resolution of the MC68328ADS LCD, many of the standard MAUI demo applications are not appropriate. The following list details the applicability of each of the standard MAUI demos:

The following applications are bit-depth independent:

- aloha
- fdraw
- gxdevcap
- hello
- inp
- msginfo
- msgrdr & msgwrtr
- sfont

The following demo applications are not supported on the MC68328ADS:

- fcopy - uses the CDi IFF file format which does not support 1 bit
- jview - does not support the low bit-depth
- auplay - no audio device
- aurecord - no audio device
- showimg - uses the CDi IFF file format which does not support 1 bit
- windraw, winink, and winmgr (a.k.a winmgrdemo) - bit depth issues

Resolved Problems

- Corrected IOBLT copy from an interlaced source
- A `gfxdev` parameter was added to `gfx_clone_vport()` to correct a problem where the cloning application would attempt to access the graphic device using the `path_id` of the application that originally created the viewport.

MAUI v2.2 Release Notes

This section contains release note information for MAUI 2.2.

Enhancements

- Color map control - these functions provide a consistent interface for MAUI application developers for color management in windowing. This includes bug fixes and enhancements for Java applications to enable transparency.

  `win_alloc_cmap_cell`

`win_alloc_cmap_colors`

`win_get_cells_params`

`win_get_cmap_params`

`win_set_color_match`

- Cursor control - these functions provide application developers the ability to programmatically move the cursor to desired screen positions plus the ability to hide and show the cursor on the fly.

`win_set_cursor_pos`

`win_set_cursor_state`

- Added new Blt and Drw functions to support overlapping copy. As a result, the graphic memory requirement for maui_win was reduced by half.

`blt_copy_oblock`

`drw_copy_oblock`

- Common VGA driver codec - this architecture allows quicker ports to new VGA devices and lower maintenance costs.
- Extended coding method definition - this supports hardware requiring large line alignments.
- Graphic driver common code enhancements to facilitate hardware BLIT acceleration and other BLIT customizations.
- Graphic hardware cursor support in GFX API and `maui_win` - these functions can result in a performance boost for platforms that support graphics hardware cursors. Hardware cursor support eliminates the significant overhead of redrawing the cursor and backgrounds beneath the cursor.

`gfx_create_cursor`

`gfx_destroy_cursor`

`gfx_get_cursor_cap`

`gfx_set_cursor`

`gfx_set_cursor_pos`

  `maui_win` - updated to take advantage of hardware cursor

- Shared library support for StrongARM, SH3, and SPARC.

Optimizations

- RAM - the dynamic footprint requirement is reduced in windowing applications. For example:

  Brutus - 320 x 240 = 76K

- Speed improvements - 10% faster in CaffeineMarks 3.0.
- Screen flicker - Corrected/reduced screen flash and cursor flicker due to extra window expose events when creating, moving and resizing windows.
- `drw_arc` - arc drawing works and is much faster

Demos

- Improved readme notes

Several new demos, including the following:

- New JPEG library viewer.
- New `gxdevcap` utility displays device capabilities information about each graphic device on the system.
- `demo_set_timeout()` now uses `signal()` instead of `intercept()` to eliminate signal handler conflicts.

Resolved Problems

- ANSI compliance/Compiler warnings
- `drw_oval_arc()` draws a correct oval
- `drw_ellipse()` in dash mode no longer strays from bounding box
- Eliminated unnecessary draws of cursor in windowing

- CLUT of top graphic device is now restored following close or restack of device
- Fixed drawing problems that were causing artifacts in the Java Gas Pump demo
- `blt_copy_block` now reads the correct number of bytes—which was causing unexplained crashes.
- `win_move_win()` no longer exposes too much area, cutting down the amount of UI refresh needed.
- The check for `topdev` is now correct for cloned devices
- Corrected identity crisis of `drw_set_context_origin()` and `drw_set_context_draw()`. Their error messages claimed they were `DRW_SET_CONTEXT_CLIP`

OS-9 Operating System Notes

This section contains release notes for the Microware OS-9 operating system.

Threads Support Release Notes

Threads Overview

In previous version of OS-9, the kernel provided support for only one thread of execution for each process. With the release of OS-9 version 3.0, the kernel supports multiple threads of execution for processes. Processes are forked with a single thread of execution, but a process can start more threads of execution during its lifetime. Each thread of execution in the process shares the memory and input/output resources of the process with all other threads of execution in that process.

Process Descriptor Changes

The OS-9 process descriptor structure as defined in the `process.h` header file has been changed to support multi-threaded processes. Previously, a process was described by a kernel process descriptor, `pr_desc` defined in `process.h`, and an I/O process descriptor, `io_proc` defined in `io.h`.

For OS-9 version 3.0, the `pr_desc` structure has been broken into two parts: one describing the thread of execution, called `pr_desc` for compatibility, and one describing the resources in use by the process, called `pr_rsrc`. These new structures are defined in `process.h`. Each process has only one `pr_desc` structure (which contains a pointer to the process' I/O process descriptor.) The process ID of the process is given by the `pr_id` field in the `pr_desc` structure. The new `process.h` header file is backward compatible with previous versions.

The new structure definitions are not visible unless `USE_V3_0_PROCDESC` is defined. If this is not defined, then the `pr_desc` structure has the format used under previous versions of

OS-9. In addition, macros to translate pre-3.0 structure member references to 3.0 structure member references are defined. This provides for system-state source code compatibility and user-state source and object code compatibility.

Kernel Changes

Five new system calls have been added to the kernel. They are _os_thfork, _os_thread, _os_thexit, _os_waitid, and _os_yield. These calls are documented in the ***Ultra C Library Reference***. A new version of the _os_gprdsc system call has been introduced.

The new _os_gprdsc call is backward compatible with OS-9 Version 2.x systems. The contents of the new split process descriptor are marshaled into the OS-9 V2.x process descriptor structure. Two new library functions have been added to obtain copies of the pr_desc and pr_rsrc structures for a thread and/or process. These calls are _os_get_prdesc and _os_getprsrc. Both of these calls use the F_GPRDSC system call code and use a system call edition of two.

The chain and exit functionality of the kernel has been changed to accommodate multi-threaded processes. The _os_chain system call has been modified for OS-9 version 3.0 to deal with chain requests from multi-threaded processes. If a process with more than one thread makes an _os_chain system call, the process exits with an EOS_THRD_INVLD status. The _os_exit system call has been modified to cause all threads of a multi-threaded process to be terminated.

System Module Changes

The ioman system module has been changed so that the pd_cproc field of path descriptors is set to the ID of the thread using the particular path. The System Security Module (SSM) has been changed to check for a process' ID by looking at the pr_id field instead of the ps_id field. This prevents the SSM image from being reloaded when switching between threads in the same process. Since all threads in a process use the same memory map, there is no need to switch the SSM image when switching between threads.

Library Changes

A new group of library calls has been implemented for OS-9 which contains most of the functionality of the POSIX pthreads group of functions. The ***Using OS-9 Threads*** manual has details about the library calls added as part of supporting threads. New versions of standard OS-9 libraries have been generated specifically for use with multi-threaded applications. These libraries are indicated with the `mt_` prefix.

Several psects within Ultra C libraries were split so that only directly referenced functions were linked to an application. Before Ultra C V2.4 for example, a reference to `malloc()` would automatically bring in `realloc()`, `calloc()`, `free()`, `_freemin()`, and `_mallocmin()`. Now the other functions are brought in only if they are referenced.

MAUI libraries ending in 'd' or 'l' have been eliminated. The library `lock.l` is new and incompatible in 3.0. Any objects that link against `lock.il/.l` should be changed to link with `sys_oscall.i/.r` and `os_lib.il/.l`.

Compiler Changes

The Ultra-C/C++ compiler front end (`cpfe`) has been changed to generate code and libraries specifically for multi-threaded applications. The `-mt` option allows code and libraries to be specified as non-threaded, inherently thread-safe, or specifically multi-threaded. The compiler warns if threaded and thread unsafe code is mixed together in an application.

Hawk Debugger Changes

The Hawk Debugger has been enhanced to facilitate the debugging of multi-threaded programs. When a process which is being debugged forks a new thread, it is treated in a manner similar to the forking of a new process. The Hawk Debugger can either start a new debug session for the new thread or can optionally let the new thread run without debug control. This feature is controlled by a menu in the **Options** dialog box selected from the **File** menu.

Precautions

The latest version of the OS-9 kernel (version 125 or newer) supports multiple threads. If you are an existing OS-9 user with a kernel version of 124 or older and you are upgrading existing systems to v3.0, please take the following precautions:

- `process.h`

  If your code accesses the process descriptor directly, define `_USE_V3_0_PROCDESC` before including `process.h` and then recompile.

- `lock.l`

  If your code accesses the `lock.l` library, you must relink to the library. The following system calls access `lock.l`:

  ```
  _os_acqlk
  _os_caqlk
  _os_crlk
  _os_ddlk
  _os_dellk
  _os_rellk
  _os_waitlk
  get_cproc
  ```

- `thread.l`

  If your code accesses the `thread.l` library, you must relink to the library. The following system calls access `thread.l`:

  ```
  kill_thread
  start_thread
  terminate_thread
  ```

- Semaphores on SuperH processors

  On SuperH processors, it was necessary to make semaphores incompatible between OS-9 version 2.X and 3.0. A semaphore initialized by 2.X object code cannot be used by code recompiled for 3.0. Semaphores initialized by OS-9 2.X code cannot be used by 3.0 code. Thus, this incompatibility is limited to semaphores that are shared by processes compiled by a mixture of 2.X and 3.0 compilers. A semaphore initialized and used exclusively by 2.X code will work correctly and a semaphore initialized and used exclusively by 3.0 code will work correctly.

Note

OS-9 v3.0 uses the Ultra C compiler v2.4. OS-9 v2.x uses earlier versions of this compiler.

User code could share semaphores without explicitly referencing them by linking in OS-9 libraries built to create applications that share semaphores. For example, 2.X MAUI windowing applications will try to share semaphores with the *maui_win* process, thus creating an incompatibility.

- Assembly language

  Hand written assembly using the `type.member` syntax is not compatible with the changes to `process.h` to support threading.

Migrating existing OS-9 code to OS-9 Version 3.0

Overview

This section covers areas to consider when migrating your code to Version 3.0 of OS-9. The first item to consider is the design goals for the threading capabilities. The following design goals were laid out before implementation began:

- Reliability is preserved
- POSIX compliant pthread API
- User and system state applications developed for previous versions of OS-9 should be binary compatible
- Process context switch time should not be affected
- Thread context switch time should be 15% - 30% faster than process context switch time
- Non-threaded application size will not increase when non-threaded libraries are used.
- Non-threaded application size will not increase more than 2% when threaded libraries are used.

With these design goals, the following items should be considered as you migrate your product to OS-9 V3.0 with threads support.

New Process Structure

As mentioned before, the structure used to define a process has changed significantly from the one used in previous versions of the operating system. It is split into two structures. One structure holds information about the process' execution context including the stack, signal and debug information (this structure is `pr_desc`). The second structure holds the process's' resource information, which includes allocated memory, linked subroutine modules, and a reference to the process's I/O descriptor (this structure is `pr_rsrc`). These definitions are in `process.h`.

The existing system call, `_os_gprdsc`, will continue to return a pre-version 3.0 format process descriptor. Thus, even if a user state process examines process descriptors, it remains source-code and binary compatible as long as it uses `_os_gprdsc()` to obtain the process descriptor.

If there is a need to examine 3.0 format thread state descriptors or process resource descriptors a pre-processor macro `_USE_V3_0_PROCDESC` and two new system calls (`_os_get_prdesc` and `_os_get_prsrc`) are provided.

Any system state code accessing a process descriptor (`pr_desc`) directly will need to be recompiled with the macro `_USE_V3_0_PROCDESC` defined. See ***Using OS-9 Threads*** for more information about the `_USE_V3_0_PRODESC` macro.

Resource Locking

The OS-9 I/O system uses resource locking calls to provide exclusive access to critical regions and help ensure proper resource management. Resource locking helps prevent data corruption by limiting process access to critical sections of code; it protects data structures from simultaneous modification by multiple processes. See the ***OS-9 Technical Manual*** for more information on resource locking. No shipping OS-9 library code references functions in `lock.l`. File managers do reference functions in `lock.l`, but are shipped in binary form only.

The locking mechanism was modified to use the process ID (PID) instead of the process descriptor pointer. Any system state module which includes `lock.h` for resource locking and is statically linked to `lock.l` will need to be relinked. It is strongly recommended that you link against `sys_oscall.r` and `os_lib.l` rather than `lock.l` for future compatibility and performance issues. On a side note, if locking is resolved in `os_lib.l` without using `lock.l`, then no recompiling or relinking is required. However, modifying the link line to include `sys_oscall.r` before `os_lib.l` will improve overall locking performance.

Semaphores

With the exception of SuperH processors, semaphores are compatible between OS-9 version 2.x and version 3.0. On SuperH processors, it was necessary to make semaphores incompatible between OS-9 version 2.X and 3.0. See **Precautions** on page 154 for more information.

User and System state applications

Non-threaded applications are binary compatible and nothing needs to change as long as the application does not reference process descriptors directly. This is true even if the process descriptor is accessed indirectly in a function call to a system library that uses the descriptor internally. All system APIs are backward compatible. Compiling a non-threaded application is exactly as it was before 3.0; options are provided to enable threading, nothing has to be done to suppress it.

User State applications should only use reentrant functions from within signal intercept routines. i.e. If a function is marked in the C Library Reference manual as non-reentrant, it must not be called from a signal interrupt routine.

File managers

No change, unless the code accesses process descriptors (`pr_desc`) directly. If so, please follow the same techniques as noted in the **New Process Structure** section above. If system resource locking is used (`_os_crlk`), please follow the same techniques as noted in the **Resource Locking** section above. The IO Manager still protects the File Manager to ensure that only one process or thread at a time is using it. Therefore, File Managers are already non-reentrant. Otherwise, it is binary compatible.

Driver code

No change, unless the code accesses process descriptors (pr_desc) directly. If so, please follow the same techniques as noted in the **New Process Structure** section above. If system resource locking is used (_os_crlk*)*, please follow the same techniques as noted in the **Resource Locking** section above. Otherwise, it is binary compatible.

Interrupt Service Routines

No change, unless the code accesses process descriptors (pr_desc) directly. If so, please follow the same techniques as noted in the **New Process Structure** section above. If system resource locking is used (_os_crlk*)*, please follow the same techniques as noted in the **Resource Locking** section above. Otherwise, it is binary compatible.

Softstax protocols

No change, unless the code accesses process descriptors (pr_desc) directly. If so, please follow the same techniques as noted in the **New Process Structure** section above. If system resource locking is used (_os_crlk*)*, please follow the same techniques as noted in the **Resource Locking** section above. Otherwise, it is binary compatible.

Allocating Memory

If a threaded application uses _srqmem or _srqcmem, it will need to include cglob.h.

OS-9 style Interprocess communication (IPC)

All existing OS-9 style IPC Methods (pipes, events, signals, data modules, semaphores) can be used by threads to communicate to threads within other processes. However, please proceed with caution when using signals by reading the usage manual.

Java/Java Native Methods Interface

No change other than you can now block on a native method without stopping the Java Virtual Machine.

Special C++ considerations

Normal caution should be used to guard against multiple threads accessing the same instance of a C++ object. When implementing C++ classes, you must lock around member functions that access globals or static data members in that instance.

Shared Library considerations

Developers must lock around global data accesses. Usually, global data is stored on a per-thread basis as thread specific data.

Naming Conventions

For backward compatibility and maximum flexibility, Microware is releasing two forms of many libraries. This allows the developer to chose between a multiple thread or single threaded strategy. The multiple thread libraries have locking around vulnerable data to protect thread collisions. The single thread library does not have this locking in place. The naming convention for these two libraries is:

- mt_<library name> * multi thread safe
- <library_name> * single threading with no locks

Threaded vs. Non threaded libraries

Since the linker verifies at link time that binaries use the correct threading library, no runtime-checking is done to ensure a threaded application binary uses a thread enabled library. Mindless mixing and matching can produce unpredictable results.

General OS-9 Version 3.0 Release Notes

Resolved Problems for Version 3.0

- CF2019: sp1577 driver bug. Error checking has been added after the do_arp function. If the entry is NULL, it will quit. Also, if there are more than 1 process waiting for an incoming call, it will return an EOS_BUSY error. Edition 14 of the driver includes the changes.
- CF2326: System state 'delete' all alarms. System call _os_salarm_delete() will not delete all alarms if set to do so. The _os_salarm_delete_sp() system call was added in order to correctly delete all active system state alarms.
- CF2537: F_CHKMEM fails for valid address and size. In certain instances F_CHKMEM would return B_EPADDR instead of SUCCESS. This has been fixed in all SSM's.
- CF2679: RBF BLKMODE transfer may incorrectly fail. Under some circumstances, a raw block mode transfer under RBF will fail with a 241 (EOS_SECT - Bad Sector Number) instead of transferring a valid block of data. This was fixed in edition #86 of RBF.
- CF7795: Need memory protection for a data module on MMU page size boundaries. A new user kernel service call, f_udatmod, has been added to force alignment of user created data modules if the SSM is present.
- CF7808: Use of semaphore can crash the system. A user state process can crash the system using semaphores. The kernel was fixed by adding semaphore queue validation code.
- CF7838: PCF Edition #68 (File modification date incorrect). PCF will not update the last modification time correctly on a file. This is fixed in edition #76 of PCF.
- CF7981: _os_ioconfig() is not implemented correctly. The _os_ioconfig() was implemented in the library as _os_ioconfig(void *param), but it was described and prototyped as _os_ioconfig(u_int32 code, void *param). The library has now been fixed.

- CF8168: llkermit included in coreboot even if not selected. The llkermit booter was included in coreboot even if not selected in the Configuration Wizard. This has been corrected.
- CF8198: Installer should not set full path to dll in registry. The Libpreload key was not being removed from the Windows registry with an un-install of Hawk. As a result, later installs of Hawk would not work correctly. As of Hawk version 2.1 and later, the Libpreload key is no longer used by Hawk.
- CF8214: mfree -e output wraps on PCAT console. The mfree utility will print extra spaces at the end of a line, causing some displays to wrap with no apparent reason. Edition #28 of mfree fixes this problem.
- CF8240: Need to add a '?' option to booters in order to display default options. This has been added to edition #19 of bootsys.
- CF8448: Rombug will stop displaying symbols after showing a very long one. This has been fixed in all current version of rombug.
- CF8621: Rombug should notify users of read-only registers. All current versions of rombug will have this functionality.
- CF8831: Implement FAT32 in PCF. Edition #77 of pcf adds this enhancement.
- CF8870: PCF on non-68k OS-9 systems does not implement _os_readln() and _os_writeln(). Edition #77 of pcf adds this functionality.
- CF8881: Under certain circumstances, PCF would return EOS_FULL even though the disk was not full. This has been corrected in edition #77 of pcf.
- CF8882: PCF may crash if SS_DISKFREE is done on an unformatted device. This has been fixed in edition #77 of pcf.
- CF8897: Under rare conditions, PCF would return bad data on a create, write, seek or read of a single file. This has been fixed in edition #77 of pcf.
- CF8930: The pcformat utility fails to properly create volumes 2GB and greater in size. Edition #4 of pcformat fixes this problem.

- CF8957: The `pcformat` utility needs to prompt the user before formatting a disk. This enhancement has been added to edition #4 of `pcformat`.
- CF9033: `SS_DSIZE` in the `cdatapi` driver for CDFM does not work on little endian processors. This has been fixed in edition #10 of `cdatapi`.
- CF9035: PCF needs 2k sector support, specifically for magneto-optical drives. This support has been added to `pcf` edition #78.
- CF9040: The `pcformat` utility may choose a bad cluster size on magneto-optical drives. This has been fixed in edition #5 of `pcformat`.
- CF9075: PCF fails to create more than 99 similar long file names in the same directory. This has been fixed in edition #79 of `pcf`.
- CF9112: Rombug needs a command to show exception short stack. All current versions of `rombug` now support the 'o' option.
- CF9159: The `termcap.h` header file contains prototype errors for certain functions in the termcap library. This has been fixed.
- CF9170: The `cdatapi` driver modifies the descriptor in memory. This bug is also in the `rbatapi` driver. Edition #11 of these drivers fixes the problem.
- CF9175: The `rbatapi` driver needs 512 byte and 1024 byte sector support. This has been added to edition #11 of `rbatapi`.
- CF9288: The `rbatapi` and `cdatapi` drivers pass an incorrect statics parameter to the ISR. This has been fixed in edition #12 of `rbatapi` and `cdatapi`.
- CF9425: The `what` utility is missing from product releases. It has now been tagged for all future releases for all processors.
- CF9524: System state applications linked against `csl` could trash the boot disk. The `csl` module now properly adjusts paths for error strings for system state applications.

- CF9738: Using FTP to move files to a PCF flash card can corrupt the card. This would occur if one were to repeatedly send the same file over and over again. This has been fixed in edition #81 of `pcf`.
- CF10012: PCF has problems calculating multi-sector writes on a FAT12 format. This bug would corrupt disks if coping over files, especially of the copying file is bigger than the original. This has been fixed in edition #83 of `pcf`.
- CF10037: PCF crashes if not inized before creating a file. This has been fixed in edition #82 of `PCF`.
- CF10269: SCSI disks hang during high data load, and disconnect during data transfer. This occurs on NCR 53C8xx and 53C7xx type SCSI controllers. The downloaded scripts used with these SCSI controllers has been updated to fix the problem. This is reflected in the latest SCSI drivers for these controllers.

Known Issues

- When `_os_waitid()` is prematurely aborted due to a signal, the signal_code pointed to by the parameter `signal` is not set to the value of the signal (nor are the parameters `*child_id` and `*status updated`). The issue is that people will expect to receive the error code `EOS_BSIG` from `_os_waitid()`, but the signal value will not be set. The signal value may be important in determining if the application should try the wait again or abort the wait process.

  The best work around for determining the signal value is to get the value of `ps_siglst` from the process/thread descriptor. That only works if you don't get another signal in the meantime. Other work-arounds may apply, but they would vary on a case-by-case basis.

OS-9 Version 2.2 Release Notes

Resolved Problems

- CF2388: The `_os_gs_popt` call could fail for an SCF device if the data buffer passed was not aligned to a 32 bit boundary. This would occur on processors that have alignment restrictions such as the PowerPC 403, SuperH, and StrongARM.
- CF2795: Both the `MWOS/SRC/DEFS/SPF/BSD/sys/types.h` and `MWOS/SRC/DEFS/UNIX/os9def.h` files defined the `caddr_t` type. These files have been modified to add a `_CADDR_T` definition which is used to prevent `caddr_t` from being defined multiple times.
- CF2848: When loading a module group (a file containing more than one module), if one or more modules in the group is already resident in memory, an `EOS_KWNMOD` error is returned and none of the modules in the group would be loaded. Previously, the system would return the error but would not free the memory used by the module group.
- CF2409: For little endian processors or big endian processors with little endian memory mapped peripherals, the byte order memory dump in RomBug is difficult to use when displaying words or longs. New `dw` and `dl` commands were added to allow dumping on word and long boundaries. The commands also allow `dws` and `dls` for dumping in non-native byte order.
- CF6504: The `bootp` client now has an adjustable number of attempts to contact a server. The default is 8, but can be set to any value from 1 to infinite using the `maxbootptry` parameter. Also, `tftp` now supports retransmitting so lost packets no longer cause the boot to fail.
- CF7593: `getwd()` did not work with the PCF file system. The `getwd()` function included in `unix.l` was updated to work with the file system type PCF.

- CF2469, 2472, 2476 2477 2478,2479 2480 2481: SCF drivers need to check WT_INPUT before sending input irq signals. The SC16450, SC7110, SC85X30, SCCD2401, SCHOST, SCRISCOM, SCSCM, SCSPU, and SC1100 SCF drivers were updated to check the SCF flag WT_INPUT prior to sending transmit wake signals from within their transmit irq routines. This prevents a case of a false wake from a transmit initiated sleep by another serial event.
- CF7358: Possible _os_chdir() failure with long paths. PCF's internal _os_chdir() function was changed to not use recursion. This fixed a possible crash situation when extremely long paths were parsed.
- CF7039: ROM boot menu prompting is unclear. The ROM's boot message now includes a more clear message when prompting for the boot menu option.
- Issue: PCF—A Memory leak in PCF makedir() was fixed.
- Issue: PCF—All driver error conditions are now preserved by PCF and returned to callers.
- Issue: PCF—A condition where PCF could send a 0 sized write to a driver was fixed.
- Issue: PCF—Problem fixed where a chd .. or chx .. would return EOS_BNAM if done from the root.
- CF4842: PCF—Now supports open and append mode.
- CF3912: PCF— _os_fd() now works on PCF.
- CF2504: PCF—SS_DELBLK notification to driver indicating when a cluster is freed was added. This is useful for Flash drivers.
- CF7718: PCF—Bug fixed where PCF could not open /md0/. or /md0/..
- Issue: PCF—PCF's internal multi-sector IO operations and caching was improved. This resulted in a large performance improvement.
- Issue: RBF—RBF's usage of CTRL_NOWRITE was corrected. This allows the disabling of access time updates on reads. The Flash-File system is the main user of this function.

OS-9 ROM-Related Problems Resolved

- Issue: bootsys—Error codes corrected when using USE_EXCPT.
- Issue: bootsys—If there is no portmenu, bootsys will use the first registered booter.
- Issue: bootsys—Code change to all booters to self-install themselves.
- Issue: exception—now handle's interrupt removal from an ISR context.
- Issue: llbootp—improved error handling in LLBOOTP.
- Issue: llip—improved error handling in llip.
- Issue: override—the override booter now can take arguments for time out length and will tolerate the absence of a configured console.
- Issue: rombreak—the rombreak booter will now insure dbgentry is installed.
- Issue: sndp—The remote system debug client now uses a compressed packet format which increases debugging speed.

SNMP Notes

This section contains release notes for the Microware implementation of SNMP.

Enhancements

- CF4931: The resident utilities, getmany, getnext, etc., now work.
- CF9586: The SNMP makefiles use `bmake` instead of `os9make`. We now ship `bmake` with all processors.
- The Emanate Lite package from SNMP Research has been added.

Known Problems

- The `target.tpl` file in `mwos/SRC/SPF/SNMP/src` must be edited before running any compiles. Only one of the supported `TARGET` = lines must be uncommented.
- All SNMP makefiles use `bmake` rather than `os9make`. To make all the subagent examples, use the appropriate makefile in the `mwos/SRC/SPF/SNMP/src/emanate/sub/examples` directory.

  For example, to build them for SNMP version 1, use the following command line:

  ```
  bmake -f Makev1.os9
  ```

  The appropriate makefile in the following directory must be run before making any example subagents:

  ```
  mwos/SRC/SPF/SNMP/src/mibs/common
  ```

OS-9 Utilities Notes

This section contains release notes for the Microware utilities.

General Notes

- All of the OS-9 utilities have been recompiled with the current UltraC compiler. This may have caused the CRC and/or size of these utilities to change even if they had no source updates. The edition number is a better indication of updates.
- The default stack size of the utilities has been increased as a result of customer requests. If your installation calls for a smaller (or larger) default stack size, use the `fixmod -us=<size> <util>` to modify the utility.

Enhancements to os9make

The following enhancements were made to the `os9make` utility.

- As of edition #97, `os9make` supports conditional statements including: `ifdef`, `ifndef`, `ifmake`, `ifnmake`, `if`, `elif`, `elifdef`, `elifndef`, `elifmake`, and `elifnmake`.
- As of edition #100, `os9make` supports use of += for incremental macro definitions.
- As of edition #132, `os9make` supports use of loops.

"for" Loops

New “for” loop functionality has been added. For example:

```
_purge: nulltrg
for NAME in $(TRGTS)
        -$(CODO) $(NAME)
        -$(DEL)  $(NAME)
        -$(CODO) $(NAME).stb
        -$(DEL)  $(NAME).stb
endfor
```

As is apparent from the above example, if you have a large number of targets, this can substantially reduce the size of your makefile. The above "for" can also be nested. For example:

```
_purge: nulltrg
for NAME in $(TRGTS)
for NAME2 in $(NAME) $(NAME).stb
          -$(CODO) $(NAME2)
          -$(DEL)  $(NAME2)
endfor
endfor
```

Macro Substitution

Be careful to not indent the contents of if and for statements. Standard make indentation rules have not changed. New macro substitution syntax can greatly simplify makefiles.

Current OS-9 makefiles require that you specify both a RELS line and an IRELS line (if both types of object are being generated). Now it is possible to specify a single SRCS line and let generic make rules handle the redundant lines.

For example, you might start with:

```
CRELS  = $(RDIR)/dodate.r $(RDIR)/doname.r $(RDIR)/domake.r \
         $(RDIR)/domac.r $(RDIR)/data.r $(RDIR)/misc.r \
         $(RDIR)/rule.r
CICODE = $(IDIR)/dodate.i $(IDIR)/doname.i $(IDIR)/domake.i \
         $(IDIR)/domac.i $(IDIR)/data.i $(IDIR)/misc.i \
         $(IDIR)/rule.i
```

This could be rewritten as follows:

```
CSRCS  = dodate.c doname.c domake.c domac.c data.c \
         misc.c rule.c
CRELS  = $(CSRCS:%.c=$(RDIR)/%.r)
CICODE = $(CSRCS:%.c=$(IDIR)/%.i)
```

Combining this with the “for” loop above, almost all redundant lines of a makefile can be eliminated. This results in a smaller, easier to read, and easier to maintain makefile.

In addition to the `% = %` modifier, there are several other new rewriting functions. There are others for performing useful operations on the source macro. For instance, to select just the last component of a file path, use the T function as follows:

```
   $(CRELS:T)
```

The result (from the example above) would be as follows:

```
   dodate.r doname.r domake.r domac.r data.r misc.r rule.r
```

Multiple modifiers may be specified and are processed in order.

Macros can be used anywhere within the modifier format strings, for even "macroized" substitution processing.

The list of macro modifiers includes the following:

E	Replace each word with its suffix		
`TEST/test.c`	`:E`		C
H	Replace each word with everything but the last component		
`TEST/test.c`	`:H`	`TEST`	
L	Lowercase each word		
`ARMv4 Mc68000`	`:u`	`armv4 mc68000`	
M	Select words matching a pattern		
`t1.c t1.h t2.c`	`:M*.c`	`t1.c`	`t2.c`
N	Select words not matching a pattern		
`t1.c t1.h t2.c`	`:N*.c`	`t1.h`	

R   Remove suffix from all words

```
     TEST/test.c t1 t2.c   :R             TEST/test      t2
```

S   Substitute old string for new string (sed syntax)

```
        testing the rest      :S/t/T/g       TesTing The resT
        testing the rest      :S/t/T/1       Testing The resT
        testing the rest      :S/t/T/        Testing The resT
        testing the rest      :S/est$/EST/   testing the rEST
        testing the rest      :S/^est/EST/   testing the rest
        testing the rest      :S/^t/T/       Testing The rest
```

U   Uppercase each word

```
        armv4 Mc6800       :u          ARMV4 MC68000
```

% = %   UNIX prefix/suffix syntax (percent is the wildcard pattern character which is left intact on the rewrite)

```
      t1.c t1.h t2.c           :%.c=%.r        t1.r t2.h t3.r
      DIR1/t1.c t2.c      :DIR1/%.c=%.r   t1.r t2.c
      t1.c t2.c             :%.c=RDIR/%.r  RDIR/t1.r RDIR/t2.r
```

Putting the macro substitution together with "for" loop processing results in a very efficient makefile.

In the generic case, you need only modify the CSRCS line and standard rules take care of the rest.

For example, based on the example above:

```
CSRCS  = dodate.c doname.c domake.c domac.c data.c \
         misc.c rule.c
CRELS  = $(CSRCS:%.c=$(RDIR)/%.r)
CICODE = $(CSRCS:%.c=$(IDIR)/%.i)

for SRC in $(CSRCS)
$(SRC:%.c=$(RDIR)/%.r): $(SRC)
        $(RCOMPILE) $(SRC)
$(SRC:%.c=$(IDIR)/%.i): $(SRC)
        $(ICOMPILE) $(SRC)
endfor
```

This example reduces a huge bulk of the makefile down to a few lines. The only line you may need to touch is the first.

New Dependency Macros

The operation in the above example is made easier and more efficient by specifying the same `$*.c` conversion in the dependency line as in the build commands which follow the line.

`$*` represents, in make syntax, to take the current object being built and strip off the directory path and suffix. Any characters following the asterisk are then appended to the new string. Therefore, `$*.c` on `RDIR/test.r` will result in `test.c`.

$@ represents the current object being built.

For example, we used to have the following:

```
$(RDIR)/dodate.r: dodate.c
        $(RCOMPILE) dodate.c
$(IDIR)/dodate.i: dodate.c
        $(ICOMPILE) dodate.c
$(RDIR)/doname.r: doname.c
        $(RCOMPILE) doname.c
$(IDIR)/doname.i: doname.c
        $(ICOMPILE) doname.c
```

for each individual source file (which could get quite lengthy). The above may also be written more efficiently as:

```
$(RDIR)/dodate.r $(IDIR)/dodate.i: dodate.c
$(RDIR)/doname.r $(IDIR)/doname.i: doname.c
$(RELS):
       $(RCOMPILE) $*.c
$(IRELS):
        $(ICOMPILE) $*.c
```

You would still require another line in the body of the makefile for each individual set of targets/source. One solution that might seem practical is shown below:

```
$(RELS): $(SRCS)
       $(RCOMPILE) $*.c
$(IRELS): $(SRCS)
        $(ICOMPILE) $*.c
```

This solution works, but it will also build every one of your targets any time a single source file is modified. This is because each individual object on the left hand side would depend upon all objects on the right side.

This is where the new syntax takes effect, as shown below:

```
$(RELS): $*.c
       $(RCOMPILE) $*.c
$(IRELS): $*.c
        $(ICOMPILE) $*.c
```

This results in dependencies that vary depending on the object whose rules are being defined. In fact, it surpasses the use of "for" in some cases. For instance, the “for” loop example from above becomes:

```
CSRCS  = dodate.c doname.c domake.c domac.c data.c \
                misc.c rule.c
CRELS  = $(CSRCS:%.c=$(RDIR)/%.r)
CICODE = $(CSRCS:%.c=$(IDIR)/%.i)
$(CRELS): $*.c
        $(RCOMPILE) $*.c
$(IRELS): $*.c
        $(ICOMPILE) $*.c
```

New Macro Assignment Modes

Following are two common OS-9 macro assignments:

```
MACRO = value1
MACRO = value2
```

With UNIX `make`, if MACRO were later expanded, it would provide a string of `value2`. With `os9make`, if MACRO were later expanded, it would provide a string of `value1`.

This is the fundamental difference (in terms of compatibility) between `os9make` and most other make programs. Each model is equally valid and useful.

Note

`os9make -l` does macro assignments in UNIX style.

`+=`	This operator will append a string to the current value of a macro in `os9make`.

For example:

```
MACRO += value2
```

If MACRO were later expanded, it would provide `value1 value2`.

`?=`	This operator has been added, which forces the assignment of a macro even if it has already been defined. In `-l` mode, this would assign the macro only if it were not previously assigned.

Therefore,

```
MACRO = value1
MACRO ?= value2
```

If MACRO were later expanded by `os9make`, it would provide a string of `value2`. If `os9make` were in `-l` mode, it would provide a string of `value1`.

`@=`	This operator allows environment variables to be set directly from a makefile. Often third party applications need environment variables set to function properly.

For example:

```
PATH @= $(MWOS)/DOS/BIN
```

Directory Building Rules

OS-9 makefiles attempt to create the RELS directory each time you perform a build. This is a function of the rule used to create the directory, which typically look as follows:

```
$(TRGT): _makdir ...
_makdir: nulltrg
        $(MAKDIR) $(RDIR)
        $(MAKDIR) $(IDIR)
nulltrg: .
         $(COMMENT)
```

In this case, nulltrg is doing nothing but adding a command to execute. Old versions of os9make required this step. Current versions of os9make do not require this step. Following is an equivalent example:

```
$(TRGT): _makdir ...
_makdir: .
        $(MAKDIR) $(RDIR)
        $(MAKDIR) $(IDIR)
```

This rule gets executed almost every time because when you modify the current directory (create a temp file for instance), the modify time on "." is updated. Since the _makdir rule depends on "." it is considered out of date. This does not make sense when directories are either there or not there. Worse since the $(TRGT) above depends upon the _makdir, it may always be relinked whenever the _makdir build commands are executed—even when the directory already exists.

The new functionality allows that if a directory (which already exists) is the object and has no dependencies, the build rules are ignored. Therefore, you can now write the following:

```
$(TRGT): $(RDIR) $(IDIR) ...
       ...
$(RDIR) $(IDIR):
        $(MAKDIR) $@
```

You might then consider a makefile that always executes makes in a set of directories. In that case, use the dependency of "." For instance:

```
TRGTS = FM DRVR
$(TRGTS): .
        $(CHD) $@; $(MAKE) -f=$(MAKENAME)
```

To conditionally call makes you can use "for" loops and "if exists" as follows:

```
TRGTS = FM DRVR
for NAME in $(TRGTS)
if exists($(NAME)/$(MAKENAME))
$(NAME): .
       $(CHD) $@; $(MAKE) -f=$(MAKENAME)
endif
endfor
```

New Dynamic Macro

In addition to the predefined macros $@, $*, and $?, the following macro is now supported:

`$<` Enumerate all the files upon which an object depends. For example:

```
foo : main.c bar.c
$(xcc) $< -f=$@
```

Other Changes

- `ifexists` and `ifnexists` are now supported for symmetry similar to `ifmake` and `ifnmak`.
- The `-n` option is now passed down through cascading makes.

Changes to Existing Utilities

Enhanced OS-9 Version 3.0

The following changes were made for the V3.0 release of OS-9.

- CF2412: The `deldir` utility has been enhanced to include support for an `-l` option. This options causes `deldir` to look for links prior to deleting directories. Use `-l` if you suspect there are links. If there are links, they will be reported and the directory will not be deleted.
- CF8214: The `-e` option in `mfree` no longer causes an unnecessary line wrap.
- CF8813: The `mdir` utility can now handle module directories with any number of modules in them.
- CF8930: The `pcformat` utility now computes a reasonable default cluster size for disks greater than 2GB in size.
- CF8957: The `pcformat` utility was enhanced to include a prompt to ensure that the user wants the disk formatted.

- CF9040: The `pcformat` utility no longer chooses a cluster size that is less than the hardware block size.
- CF9906: `os9make` can how handle the case when an "`if`" clause is terminated by an "`ifmake`" or "`ifnmake`".
- CF10108: `idbgen`, edition #239, now supports all of the PowerPC targets currently supported by the Ultra C/C++ compiler.

Enhanced OS-9 V2.0

The following changes were made for the V2.0 release of OS-9.

- `copy`

  New `-c` option to convert end of line characters between different line terminations.
- `dir`

  Fixed command line argument parsing, year and file size print, and sector number sign.
- `events`

  Added `-h` and `-k` options.
- `format`

  New prompts for cluster size and number of sectors, new support for partitioning large disks, and corrected problem error checking auto-sized disks.
- `kermit`

  Fixed several problems that could cause memory corruption.
- `login`

  Fixed to use environment to set shell. Now works without a current directory (for disk-less systems).
- `mshell`

  Fixed several problems that could cause memory corruption. Added `-<sig>` option to kill command.

- paths

  Can now display more than 1024 processes.

- procs

  New command line argument to specify individual process ids.

- save

  Now supports the use of -f and -z together.

- shell

  Fixed several problems that could cause memory corruption. Added -<sig> option to kill command.

- sysgo

  Now uses mshell for processing the startup file and as the shell forked on the console port for user interaction. The standard sysgo now looks for the startup file in the SYS subdirectory of the default disk. This provides consistency with other versions of OS-9 and helps with the transition to startup files, which can take advantage of mshell specific features.

  Sysgo_nodisk also uses mshell. Sysgo_shell has been added as a backward compatible version of sysgo that still forks shell and looks for the startup file in the root of the initial disk device.

- tsmon

  Fixes to use environment to set shell.

  No longer bus traps when using the -z (from stdin) option.

- umacs

  Fixed several problems that could cause memory corruption.

- unlink

  The -f option can no longer enter an infinite loop by attempting to unlink members of a module group.

- xmode

  No longer bus traps when using the -z (from stdin) option or when attempting to change a write protected descriptor.

Product Discrepancy Report

To: Microware Customer Support

FAX: 515-224-1352

From:____

Company:____

Phone:____

Fax:______________________Email:___________________

Product Name:

Description of Problem:

____

____

____

____

____

____

____

____

____

____

Host Platform____

Target Platform_______________________________________