Microware OS-9 Release Notes

Version 4.1

www.radisys.com

Revision A • May 2003

Copyright and publication information

This manual reflects version 4.1 of Microware OS-9. 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 Corporation.

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.

May 2003

Contents

1 Introduction

Microware OS-9 version 4.1 represents a maintenance and update release to incorporate all of the improvements that have been introduced into the component parts.

This release only includes updates for the SH-4 and ARM processors. For information on the latest enhancements to Microware OS-9 for MIPS, PowerPC, SH-3, SH5, or x86 boards, refer to *Microware OS-9 Release Notes version 3.2*. For the latest enhancements to Microware OS-9 for IXP1200, refer to *Microware OS-9 Release Notes version 4.0*.

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

In addition, it is recommended that you read the release notes from your previous version of OS-9 in conjunction with the notes for the current release.

The following list describes the major enhancements that were made to OS-9 for this release:

- Support for 64-bit Integers

  Support for the `long long` data type as a 64-bit integer was added to the Ultra C/C++ Compiler. This was done to support existing software (such as IP version 6 and third-party applications) that assumes 64-bit integer operations are allowed.
- Updated Network Addressing

  The TCP/IP stack now supports both IP version 4 and IP version 6.

2 Operating System

This chapter provides an overview of the changes and improvements made to OS-9 since the last release. The following sections are included:

Enhancements

The following section describes enhancements that were made to the operating system since the last release. Where applicable, the Customer First (CF) incident number is listed.

- CF 8814: The C libraries for OS-9 processors have been enhanced to include an `_os_sema_tryp()` function. This function attempts to claim a semaphore, but returns an error if it cannot immediately be claimed.
- CF 13012: New low-level console service, `rom_fprintf`.

  The console services in the low-level ROM structure now has a function pointed to by the `rom_fprintf` field. Previously, the value of `rom_fprintf` was NULL.
- CF 13348: ModMan File Manager components.

  ModMan and `mm` descriptor have been added for all processors.
- CF 13747: `btfuncs.a` has been updated to detect the 7450 processor. The CPU table has also been synchronized to 16 entries. These updates included in edition 45 or greater of `btfuncs.a`.
- CF 13751: Detection of the MVME2400 and 5100 to the HAWK chipset choices has been added.
- CF 13753: `epic.c` has been updated to use EUMBARR_VALUE instead of getting its base address information from the PCI Bus. This can be used for basic hardware detection, if the EPIC is responding. In addition, EOS_HARDWARE is now returned if EUMBARR is zero. These updates included in edition 3 or greater of `epic.c`.
- CF 13835: A new class of semaphores (usemaphores) was introduced into OS-9. These semaphores automatically unlock and unlink when the linked process terminates. Contact Microware Customer Support to get the libraries containing these new calls for ARMv4 big-endian.

Resolved Problems

The following section gives a description of Customer First incidents related to the OS-9 operating system and how they were resolved for the current release.

- CF 10231: The `setpr` system call does not always cause a process reschedule to occur.

Previously, if a currently running process were to set its priority low, a reschedule would not occur to allow a higher priority process to run immediately. This was fixed in Edition 151 of the kernel.

- CF 10489: `siuirq.c` and `cpicirq.c` routines.

  Originally, only the `cpicirq.c` attempted to re-enable interrupts to allow nesting. This allowed all non-CPM interrupt sources to preempt and interrupt the CPM interrupt sources. Code in `cpicirq.c` was modified so that after the appropriate masking is done on the CPM module, the CPM interrupt to the SIU is re-enabled.

- CF 12343: Compressed/ZIP booter.

  OS-9 booters now support compression of boot files. The compression used is `zlib`, which is the same algorithm used by `gzip` and `pkzip`. Refer to `MWOS/SRC/LIB/ZLIB-1.1.4/zlib.html` or `www.gzip.org/zlib` for more information on `zlib`. The utility is `mbc.exe`.

- CF 12896: The I2C library code does not pass interrupt.

  This issue concerns the I2C library code for the 8xx PowerPC processor. The interrupt service routine did not properly check for the interrupt cause and return EOS_NOTME if the cause was inappropriate. This problem has been fixed.

- CF 13002: The file `SC8250.a` contains an IRQ and stack error-related error.

  In between version 1 and 2 of this file, instead of pushing and popping `sr` onto the stack, the file saved `sr` in register d6. The code has been changed to correctly use register d6.

- CF 13023: Setting `d_minpty` with low priority process in active queue hangs system.

  The OS-9 kernel no longer masks interrupts when only non-runnable programs appear in the active queue.

- CF 13158: The `d_ticks` rollover concept of using the maximum tick count for alarm/sleep has been removed.

  The kernel has been updated for `d_ticks` rollover. In addition, the kernel no longer has a concept of a maximum (undocumented) tick count for an alarm or sleep. These issues have been resolved in edition 154 or greater of the kernel.

- CF 13162: Exiting thread can hang application.

IOMAN's I_SETSTAT parameter bock was not being setup correctly for thread termination, which caused threaded applications to hang. IOMAN Edition 35 or greater resolves this problem.

- CF 13321: Timer Tick (`tk1100.c`) is not running with the desired precision on ARM.

  The timer lost between six to 30 seconds of time every hour. This has been fixed as of Edition 6 of `tk1100`.

- CF 13328: Bad address passed to the exception handler.

  OS-9 kernels were fixed so that the exception handler can have access to both the address of the instruction that caused the error and the address that caused the exception (if applicable).

- CF 13398: SCF: When `_os_readln()` is used on a path that is configured with a timeout value greater than one, an infinite loop is performed.

  The check for timeout on sleep code was positioned such that the conditions were never true and the timeout was not reset after each byte was received. This problem has been resolved in SCF Edition 31 or greater.

- CF 13664: `_os_sigreset()` does not work on ARM processors.

  The OS-9 kernel was fixed so that `_os_sigreset()` can work correctly on ARM processors.

- CF 13660: Ctrl-C/Ctrl-E is not delivered to correct process when `mt_csl` utilized.

  `ioman` was enhanced to know which process was the last to use a port. This ensures that Ctrl-C and Ctrl-E are sent to the correct process.

- CF 13713: The PC keyboard stops working after the LEDs are switched on or off.

  The keyboard driver sends a two-byte sequence to the keyboard to switch the LED (Num-Lock) on and off. This produces an answer from the keyboard. In some cases, however, the answer may be read, but an interrupt still occurs. In the interrupt function of the driver, the ready bit (SR_RxRDY) is checked. Because the LED light was turned off, no character could be read. Code has been added to resolve this issue.

- CF 13707: The x86 kernel does not handle `uninit` trap handlers.

  Several aspects of trap handler in the OS-9/x86 kernel were fixed.

- CF 13746: An incorrect ROM search is performed if multiple entries exist in `MWOS/OS9000/SRC/ROM/boot.c`.

  Incorrect ROM ranges were searched if more than one ROM entry existed in the memlist. This problem has been resolved in Edition 66 or greater of `boot.c`.
- CF 13748: SCSI driver NCR8xx is now detecting the 895A correctly.

  Unlike all previous chips, the 895A has a 256-byte read and write FIFO. In addition, the PCI Device ID search has been updated for use with the 895 and 895A.
- CF 13749: NCR8xx low-level driver add-on for 895A.

  Three modifications were made to the NCR8 low-level driver for 895A:

  1. The reset time has been changed from 20 µsec to 350 msec.
  2. Support for 895A has been added.
  3. The chip now resets at termination.
- CF 13752: `MWOS/SRC/DEFS/hw/16450.h` has generic set of OSET.

  The MVME5100 has a register offset of $10 so it needs OSET defined in the makefile.
- CF 13754: IDE low-level driver updates.

  Some new IDE hardware has a new class code (0x0101fa) in the PCI information. The driver now correctly finds the new IDE PCI IDs.
- CF 13755: `sccpm` modifications for PPC850/823.

  Definitions for the PowerPC 850 and 823 boards were added where the SMC2 is on PA8/9.
- CF 13757: Set the IRQ Mask in the SIU and re-enable the higher IRQs.

  The handling has been updated similarly to `siuirq.c` .
- CF 13781: The `rom` module service `goodmodule()` fails when the boot size equals the module size.

  The ROM software was fixed so that if the last module in a ROM-based boot ends on the last byte of the ROM memory, it is still accepted as a valid module.
- CF 13803: The low-level 16450/16550 driver needed baud rates higher than 38,400

The low-level 16450/16550 serial driver now supports baud rates higher than 38,400 up to 115,200. This allows the higher baud rates to be selected in the Wizard low-level serial port configuration.

- CF 13816: File creation results in empty file instead of 000:248 error on full disk.

  The minimum allocation unit is one cluster (cluster size set at format). As such, one cluster is required for directory or file creation. PCF edition 89 or greater has been updated to apply the check for free clusters to both files and directories.

- CF 13923: `_os_salarm_reset()` does not work correctly with cyclic alarms.

  The OS-9 kernel was fixed so that `_os_salarm_reset()` could be used with cyclic alarms. `_os_salarm_reset()` now resets both the next alarm and all subsequent alarms. It can also be used to change all attributes of the alarm.

- CF 14002: MBX8xx receives data exceptions when running simple script.

  The PowerPC 8xx SSM modules were fixed to correctly maintain ASID ownership when a large number of threaded processes exist simultaneously.

- CF 14021: Wrong descriptor generated for `k0_fr`, `k0_gr`,

  The proper descriptor is now generated.

- CF 14269 OS-9 can leave a near-dead process sitting around

  The kernel was fixed so that if a thread calls _os_exit() after the main thread has already terminated, the main thread will appear suitably dead for its parent process and both the process descriptor for the child thread and the main thread will be returned to the free memory pool.

- CF 14472 process can't be waited on if it's an orphan thread descriptor

  The kernel was fixed so that a parent process could successfully call _os_waitid() for its child process even if the main thread of the child process was an orphan thread, but not orphan process.

3 Processor-Specific Notes

This chapter provides an overview of processor-specific enhancements and modifications for Microware OS-9 version 4.1.

The following sections are included in this chapter:

SH-4 Notes

The following sections represent changes and updates to OS-9 for SH-4 version 4.1.

Enhancements

This section describes enhancements made to OS-9 for SH-4 since the last release.

- CF 14670: SH7750SE port doesn't have Hawkeye/Profiler support

  Hawkeye and Profiler support has been added to the Configuration Wizard.
- CF 14778: SH7750RSE files need to be shipped

  The SH7750RSE board port has been added.
- CF 14607: SH7750SE Wizard doesn't have compressed bootfile functionality

  Compression booting has been added for the SH7750SE in the Configuration Wizard.

Resolved Problems

This section provides a list of SH-4-specific Customer First (CF) incidents and how they were resolved for the current release.

- CF 13367 illegal BCD YEAR, rtc7750 & rtc7708

  The real-time clock drivers for the SH-3 and SH-4 families of processors have been fixed to properly represent the current time in the RTC registers in BCD. Previously, illegal BCD values were being placed into the registers which could cause problems in "rollover" years (2009, 2019, etc.)
- CF 14656: IGS CyberPro 5050 drivers

  The IGS CyberPro 5050 drivers have been removed.

Known Issues

This section describes the known issues that exist for this release of OS-9 for SH-4. Workarounds are provided, where possible.

- CF 14723: USB Host was not tested for SH4

Due to hardware limitations, the software for the USB Host addon for SH4 was not tested before the CD was released.

ARM/StrongARM Notes

The following sections represent changes and updates to OS-9 for ARM/StrongARM version 4.1.

Enhancements

This section describes enhancements made to OS-9 for ARM/StrongARM since the last release.

- CF 14863: Shipped coreboot image and uncompress

  Bootfile uncompression capability has been added to the default shipped coreboot images for all ARM ports except PID7T.
- CF 14913: Broadcast bit doesn't get set for sp8900 driver

  Set the broadcast bit in `init()` in `entry.c` is now set as of edition #4 of sp8900 for both CDB89712 and EP712.
- CF 14914: sp8900 driver does not have multicast support

  Added support for accepting multicast packets in entry.c, init.c, sp8900.h and defs.h in edition #4 of sp8900 for both CDB89712 and EP712.

Resolved Problems

This section provides a list of ARM/StrongARM-specific Customer First incidents and how they were resolved for the current release.

- CF 14730: dhcp hangs system if it already has a valid address

  When using DHCP on the GraphicsClient, the low level ethernet address must be set to 0.0.0.0 or the low level ethernet driver should not be included into the coreboot. If a valid IP address is set for the low level ethernet address, then when booting the following error will appear:

  ```
  DHCP: IP address already exists <low level ethernet address>
  to get new IP address: dhcp <eth_dev> -override
  ```

  at this point the target will hang and nothing can be done.
- CF 14811: pscf5 isn't needed in lan.ml for GraphicsClient

pscf5 has been commented out in both lan.ml and spf.ml for the GraphicsClient.

- CF 14906: RPC demos removed between SARM v3.2 and SARM v4.0

  RPC demos available in previous versions of OS-9 are no longer supported and were not shipped in this release.

Known Issues

This section describes the known issues that exist for this release of OS-9 for ARM/StrongARM. Workarounds are provided, where possible.

- BLS for ARM Prospector has not been tested

  Due to hardware limitations, the software for the ARM Prospector was not fully tested before the CD was released.
- CF 14799: The Assabet receives ethernet traffic very slowly

  During testing, when the Assabet was receiving lots of information it was very slow (less than 100k/sec).
- CF 14849: PC Card 3Com Etherlink III does not work on Assabet

  We recommend using the onboard Ethernet device.
- CF 14851: The LP-E PC Card does not work on Assabet

  We recommend using the onboard Ethernet device.
- CF 14853: NFS client does not work on Assabet

  Copying files to the mounted drive causes the system to lock up.

4 Host Applications

This chapter contains release notes for host applications used with OS-9 version 4.1. It includes the following sections:

Configuration Wizard Notes

The following sections represent changes and updates for the Configuration Wizard.

Enhancements

The following list describes the general enhancements made to the Configuration Wizard since the last release. Where applicable, Customer First (CF) incidents are included.

- SoftStax is now automatically enabled.
- You can now start `inetd` or individual networking daemons.
- The **Advanced Inetd Options** dialog allows you to select custom `inetd.conf` file instead of using the Wizard-generated `inetd.conf` to build the bootfile.
- You can now configure multiple Ethernet interfaces in the Configuration Wizard. Select the `Commit Change` button to save changes to an interface.
- Check boxes for IPv4 and IPv6 allow you to select which address format(s) you prefer to use for an interface.
- An **Export Interface Report** button has been added in the Interface Configuration dialog, enabling you to save your configuration information to a text file for later reference.
- The Wizard will accept IPv4 and IPv6-format addresses for DNS Sever Search Order and gateway addresses.
- `lan.ml` is the IPv6 equivalent of `spf.ml`; it lists modules to be included in your bootfile. All ports have been updated so that the Wizard looks for `lan.ml` when you include SoftStax support in your boot.
- CF 9703: Cannot set two different IP addresses in OS-9 Wizard.

  A new multiple-interface Ethernet card configuration allows you to set multiple IP addresses for one machine.
- CF 11654: IP and `llbootp` update.

  Boots created with the Wizard may now obtain high-level IP address information from information provided to `bootp`. This new feature must be configured in the port's `.ini` file.
- CF 12388: The Wizard is hardcoded to look at the `.ml` files when macros exist.

The filenames `lan.ml`, `spf.ml`, `user.ml`, `bootfile.ml`, and `system_soft.ml` are no longer hardcoded into the Wizard. You can define module list filenames and absolute and relative pathnames in your port's `.ini` file.

- CF 12790: Use `.ini` file to initialize network settings.

  A button has been added in the **DNS Configuration** dialog that allows you to import DNS information from any existing `.ini` file into the current configuration.

- CF 12863: Add pause on/off check box.

  A check box has been added to override PAGEPAUSE included in the **Define Term/Port** dialog.

- CF 12918: Add time to the date placed generated files.

  Wizard-generated configuration files now include the time of creation in the file header.

- CF 13048: If `mbinstall` is not checked in, the init string has options `-m=2048k`.

  The initial parameter string no longer includes `mbinstall` when it is not selected via **Init Options** -> **SoftStax Setup** tab.

- CF 13550: RomBug in the bootfile check box should also include `dbgentry`, `dbgserv`.

  `dbgserv` and `dbgentry` will be included in bootfile if the **Rombug in bootfile** box is checked and no debugger method is selected in the coreboot.

- CF 13601: Support user parameters that can be applied to any configuration.

  An advanced feature has been added that allows you to append custom parameters to the beginning or ending of any string of initialization parameters automatically generated by the Wizard. This feature is available when any initial disk other than "User" is selected.

- CF 13824: Add more check boxes to **SPF Options** dialog.

  The **SoftStax Options** dialog now contains up to 75 check boxes for modules included in bootfile. Options must be configured in port `.ini` file and `lan.ml`, `spf.ml`, or equivalent.

- CF 14876: Add cdbval and msginfo to MAUI demos

  Added cdbval and msginfo to maui.ml for all ARM ports.

- CF 14901: Coreboot->Main Configuration needs Compression tab

Added "Bootfile Compression" options tab in the coreboot configuration in the Configuration Wizard. Fix is found in v2.34 ed #128 and higher of Configuration Wizard.

Resolved Problems

This section provides a list of Customer First incidents related to the Configuration Wizard and how they were resolved for the current release.

- CF 11191: The configuration is lost when a new version is installed.

  The Wizard now uses `os9p.ini` for user-saved configuration files. Port information is stored in separate files in `/MWOS/DOS/BIN/WIZARD`. This prevents user-saved configurations from being lost when new versions of OS-9 are installed.
- CF 12387: Include `hlproto`, `undpd`, and `undpdc` in the boot.

  The Wizard now includes user-state debugging modules when SPF is disabled or high-level Ethernet is disabled.
- CF 12878: DNS Configuration problem in the Wizard.

  Domain names and DNS suffixes may now contain 256 characters.
- CF 12902: SLIP-DNS hostname problem in the Wizard.

  When there are no Ethernet interfaces enabled, the host file created by the Wizard will associate the SLIP or PPP connection with the hostname entered by the user.
- CF 13788: The Wizard does not report an error unless the environment is properly set.

  If the host system environment does not have correct `MWOS` variable, the Wizard will create a coreboot with a size of zero. The Wizard has been modified to alert you to check the environment when an empty coreboot is built.
- CF 13892: The Wizard crashes if it resides in a directory other than `MWOS/DOS/BIN`.

  The Wizard now receives the location of the `MWOS` directory from the host system environment instead of from the location of `os9p.exe`.
- CF 14032: Changing the configuration name does not save the correct information.

  The Wizard will now save correct user-configuration information to `os9p.exe` when you select `File -> Change Configuration Name`.

- CF 14916: Bootfile only/low-level debugging modules

  It is now possible to include low-level ethernet debugger files in a Configuration Wizard bootfile build without having to configure the coreboot. Check the "User State Debugging" box on the Wizard build screen.

Known Issues

This section describes the known issues that exist for this release of the Configuration Wizard. Workarounds are provided, where possible.

- Low-level IP information must be entered in IPv4 format.
- If DHCP is used and the gateway is entered, the Wizard name resolution does not work.

  There are two workarounds for this issue:

  1. Do not enter the default gateway in the Wizard when using DHCP. DHCP server should return default gateway.
  2. If your DHCP server does not return a default gateway, set it in the Wizard, then delete the default route and add it again manually, as shown below:

     ```
     $route delete default
     $route add default <default_gateway>
     ```

Hawk Notes

The following sections represent changes and updates to Hawk since the last release.

Enhancements

The following list describes the general enhancements made to Hawk for the current release. Where applicable, Customer First incidents are included.

- The Differencing feature has been enhanced. A new Differencing submenu provides access to differencing features, including text files, binary files, and directories.
- The Show References to (selected symbol) dialog and Show Symbols Called or Used by (Function or Class Name) dialogs now provide information on how C, C++, Java or C# symbols are used in your project.

- Regular expression support is available for both strings and numbers, allowing you to more easily identify which code is ChromaCoded.
- CodeSense has been enhanced as follows:
  1. You can select the Advanced Options tab on the Customize -> CodeSense Global Configuration dialog to access several settings for enhancing the functionality of CodeSense.
  2. A Maximum Cache Size slider on the CodeSense Advanced Options dialog has been added. This feature allows you to change the number of hits from CodeSense lookups that are retained in memory.
  3. Mark the option Filter Matches by File Type on the Customize -> Language -> CodeSense dialog to filter lookups for the file type(s) selected on the Language dialog. To see CodeSense information that applies to any file type, disable this option.
  4. CodeSense now fully supports Java packages and C++/C# namespaces.
  5. You can now specify which parser language to use for a given file type or library database.
- Documents opened in Hawk are now accessible from a tab bar across the top of the Edit window.
- Three new check boxes are available from the Customize -> Environment -> General tab:
  1. Enable tab virtual space
  2. Enable EOL virtual space
  3. Standard MDI presentation
- The File View window (now known as the Project tab of the Project Window) displays the current project space with all of its member projects contained underneath. Each project can be further expanded to display the member files in hierarchical form.
- The File –> New (Create New Document) dialog allows you to start your editing session by populating a new document with one or more code snippets from the CodeFolio directory. Include file headers, function headers, and more to eliminate repetitive typing, and assure consistent code formatting.
- Microsoft Visual Studio.NET projects (.VCPROJ, .VBPROJ and.CSPROJ) and solutions (.SLN) can now be imported into Hawk.

- The underlying functions for the CUA, BRIEF, vi and Epsilon keymaps are now documented online to illustrate their functionality, parameters, return values and keyboard shortcuts, when applicable.
- Both named and unnamed bookmarks can now be saved in the Bookmark Database when a file is closed, either during a session or at the end.
- CF 9436: Add client installation.

  Client Installation is now available. Hawk will continue to use the path specified in the `MWOS` environment variable to find all program files and the rest of OS-9. If it exists, the path specified in a new environment variable named HAWKCLIENT will be used for all user configuration files. If any of the user configuration files do not exist in the HAWKCLIENT path, they will be opened on the `MWOS` path. The product installers have an additional setup type called, "Install Client", which functions similarly to the other options, but displays an additional dialog prompting for the path to the server directory. The only files installed with this option will be the user configuration files installed to the path chosen for HAWKCLIENT.
- CF 12712: Delete files out of dialog.

  A handler has been added to the Directory Selection dialog so the Delete key can be used to delete a folder from the list. Once a folder has been deleted, the folder above it will be highlighted.
- CF 12735: C++ support for `hawk_control_log` in `slmlib.h`.

  If the log file is included in a `.cpp` file, the `hawk_contol_log` function will have "C" { } code around it.
- CF 13051: Load all modules.

  An item has been added to a component's Property dialog and pop-up menu called Exclude from Load All. If you select this item, the component will be skipped when the Load All item is executed.
- CF 13084: Enhancements for Hawk's `build.log` file.

  The `build.log` file is now saved in the project directory. In the log, the “\r” character that was used for a new line in the command line memo has been replaced with a normal line feed.
- CF 13085: The `hawkdbgr.exe` should have a `-z` option to load multiple files.

  Code was added so that a file can be used to input a list of modules to load.
- CF 13219: Debug parallel processes in the Hawk Debugger.

Hawk will now start a stand alone debugger if the Hawk IDE is currently debugging a process and Debug -> Connect is selected.

- CF 13263: Display of data types in Hawk.

  If you right-click on a value in the Locals or Watch window, you can now select two more menu items (Display Decimal and Display Hexadecimal); each can be used to change the display type of a single entry in either grid.

- CF 13603: Add 64-bit support for Hawk.

  Code was added so that Hawk can display 64-bit values correctly in the Locals and Watch windows.

- CF 13760: The Hawk Debugger Connect to module pull-down menu keeps no record of previous modules.

  Code was added so that eight entries are saved in the Attach to module edit box.

- CF 14047: Specify multiple module to attach in Hawk Debugger.

  The `-AM` command line option was changed to allow the input of a list of modules.

Resolved Problems

This section provides a list of Hawk-specific Customer First incidents and how they were resolved for the current release.

- CF 11187: Hawk cannot launch the Profiler if the `MWOS` environment variable contains spaces.

  Quotes have been added around the string that contains the path to the java `.jar` file that is executed to run the Profiler.

- CF 11851: System-state Hawk needs the ability to boot a system in stages.

  A “go boot” option was added to the Debug menu that has the same functionality as typing in `gb` when booting a system in ROMBug.

- CF 12208: When you are debugging, if you perform a step-through command in Hawk using the `int*float` operation, it will result an incorrect answer.

  The bug was included with the daemon `spfndpdc`. It was not properly restoring floating point registers when doing a `step`, `next`, `go`, etc. This has been corrected with the current `spfndpdc` module.

- CF 12344: Cannot set breakpoint in module source located in a directory with special characters.

  The `mwsrcdbg.dll` previously used many unix conventions and characters like '+' were not allowed in a pathname. The function that parses the breakpoint command has been updated to allow "+", ".", and other special characters in the pathname.

- CF 12740: Corrupt source code folder.

  Previously, the default size of the text box was 256. This would cause any source paths that were set over the maximum character length to be deleted. The folder is now set to have a maximum of 4000 characters.

- CF 12854: After switching from user-state to system-state, the processor selection changes.

  The processor selection for a component is now retained if the component type is switched.

- CF 12856: Problem setting break points.

  Previously, breakpoints were both set and stored in the same manner. For example, if a user typed a function name and then tried to delete the breakpoint by selecting it the left side of the source window, the filename and line number would not match any of the set breakpoints; consequently, the breakpoint would be deleted. Hawk has been updated so that it will remember both the function name and the filename and line number of the breakpoint.

- CF 12933: The Rebuild all projects item does not work.

  Previously, if a project had more than one component when the Rebuild all projects action was performed, Hawk attempted to advance to the next project before all of the first project's components had completed building, thus causing access violations. The problem has been corrected.

- CF 12942: The Hawk Debugger cannot initiate a connection if a parameter includes "@" character.

  The function used to parse the string passed to `mwsrcdbg.dll` during a program fork has been rewritten to allow the @ character to be in the environment variables or command parameters.

- CF 13153: Deleting a project in Hawk causes an access violation.

  Hawk looked at the function to delete a component when the Delete key was pressed while a project was highlighted. The delete function now checks if the highlighted item is a component or a project and then calls the correct routine to delete the selected item.

- CF 13196: Edit memory in Locals window.

  Previously, the function used to edit memory inserted an extra two bytes of data. The function has been updated so that it writes the correct amount of data; thus, modifying a “short” will not change the value of a variable stored right after it.
- CF 13209: Hawk Process I/O window does not remain scrolled at the bottom.

  There is a property of the control TAdTerminal called “ClientOriginRow”. If you set this value to 1, the terminal window will automatically scroll down so that the top line of the terminal window is visible, regardless of where you set the scrollbar.
- CF 13210: Hawk Process I/O window adds extra carriage returns.

  Previously, Hawk added an extra carriage return each time the Return key was pressed in this window. This problem has been fixed.
- CF 13337: The underscore character cannot “prefix” a renamed component.

  When you select the Rename item by right-clicking on a component, you can now also select component names that begin with an underscore.
- CF 13338: Source files of type `*.a` are not visible in the Create New Component (Unit) dialog.

  The text of the file filter displayed type `*.a`; however, that item was missing from the actual filter component. It has been added; thus, assembly files are displayed with the other source files.
- CF 13505: Hawk displays an error message after the current project is changed.

  Previously, switching between two projects in a project space several times caused an access violation. This problem has been corrected.
- CF 13404: Hawk cannot debug multi-threaded applications.

  The x86-specific code in `mwsrcdbg.dll` was not reading the pointer to the thread function correctly; this caused an error when the new debugger was launched. This problem has been fixed.
- CF 13469: Hawk does not build modified source code.

  Previously, the method used to determine whether or not a file needed to be rebuilt did not work. A new method has been implemented.
- CF 13493: Hawk is appending `mwhawk.pst` to open source files.

This was a problem in CodeWright 6.6 that was fixed with version 7.0. CodeWright 7.0 has been integrated into this Hawk 2.3.

- CF 13594: Selecting the `.cs` file type in the Customize -> Environment dialog causes Hawk to hang.

  Code was added to `cwstart.dll` to support the `.cs` type in the Customize -> Language dialog.

- CF 13812: The Hawk Debugger cannot find the current file.

  Previously, Hawk would read the symbol file as it loaded a module and then move to main. If the source path was not set correctly, it would prompt the user to find the file. This allowed the file to display, but setting breakpoints on the file did not work because the symbol information was not updated with the correct path.

  A dialog has been added in the latest version of `mwsrcdbg.dll` that prompts you to find the file as the symbol information is read. The path to the source path will then be added to the source path debug options so you will not be prompted to find other source files in the same directory. Thus, breakpoints can now be set, as the path information for the symbol now matches the file's path.

- CF 13860: It is not possible to browse file paths in the Project Properties dialog.

  The version of the component used to browse through the folders was not supported under Windows XP. Hawk has been rebuilt with an updated version of the component; this error no longer occurs.

- CF 14770: Hawk problems when syntax errors are in source code

  The Microware error parsing functions were missing from a previous release. This would cause the functions that show where a certain error occurred to not work. The error parsing function has been added back in version 2.3.4.0 of hawkprj.dll.

- CF 14865/14894: If a connection is lost, hawk won't allow another right-click debug

  There is a flag that is set when a component is being debugged that is used to make sure the debugger is started twice for the same component. This flag was not being reset to false when a connection with the target was lost. This would result in the inability to debug a component again if the connection with the target was lost for some reason. This has been fixed in version 2.3.1.29 of hawkdbg.dll.

- CF 14866: the "ident module" function doesn't work in Hawk.

The "ident module" menu item in Hawk now works correctly. Previously, incorrect values were reported or Hawk would lock up in an infinite loop.

- CF 14867: When editing register values, there's no cursor

  The row width in the registry window was too small to allow for a cursor to be drawn. The width has been increased and the cursor now fits in the row. Because the display of the registers window is always in hexadecimal format the editing of the register values has been changed so that it is also always in hexadecimal format.

- CF 14870/14675: Hawk standalone debugger fails to load module (system-state)

  undpdc and spfndpc weren't exiting correctly when a module was loaded. This was causing an access violation in Hawk when a module was loaded. The modules have been fixed in edition 72 and the access violation no longer occurs.

- CF 14758: Forking a module in Hawk doesn't handle parameters properly.

  Hawk was not parsing the command correctly to get just the name of the module being forked if command line parameters were used. The effect of this was Hawk was unable to find the correct .dbg and .stb files because it was looking for the wrong things. This has been fixed in version 2.3.4.2 of mwsrcdbg.dll.

- CF 14894: "This component is currently being debugged" error when not debugging.

  There is a flag that is set when a component is being debugged that is used to make sure the debugger is started twice for the same component. This flag was not being reset to false when a connection with the target was lost. This would result in the inability to debug a component again if the connection with the target was lost for some reason. This has been fixed in version 2.3.1.29 of hawkdbg.dll.

HawkEye Notes

Resolved Problems

This section provides a list of HawkEye-specific Customer First incidents and how they were resolved for the current release.

- CF 13315: In HawkEye, the Chart information differs for Windows NT.

  The action on an item in the Elapsed Time Chart now behaves correctly.

5 Components

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

OS-9 Compiler Notes

The following sections contain release notes for the OS-9 Compiler.

Enhancements

The following processors now support 64-bit integer operations using the non-standard data type `long long`:

- ARMv4 (big- and little-endian)
- SH-3/SH-4
- PowerPC
- SH-5
- x86/Pentium
- MIPS64

Resolved Problems

This section gives a description of Customer First (CF) incidents related to the Compiler and how they were resolved for the current release.

- CF 8178: The ARM back-end performs incorrect data area layout, which results in linker errors.

  The ARM back-ends were fixed to recognize the fact that there are three different lengths of data area references for performing data area layout. This eliminates said linker errors.
- CF 11223: The `xcc.exe` file does not compute the correct ROF name in C89 mode.

  Previously, the executive computed an incorrect ROF file name. It now computes the correct ROF file name.
- CF 11607: The `-qi` option for `xcc.exe` does not work.

  The executive no longer puts `implicit_include` on all front-end command lines for C++ source files. Instead, it now only places it on the front-end command line if `-qi` is used.
- CF 12706, 12709: MIPS `const` qualified pointer is incorrect.

  Previously, on MIPS processors, the `const` qualified pointer to `const` qualified objects (where the destination of the pointer was offset `0x7ff0` from the base of the module) was incorrectly interpreted as NULL. This problem was corrected; the front-end has been modified to initialize NULL `const` qualified pointers with –

0x8000. This ensures that NULL const qualified pointers are unambiguous.

- CF 12759: _os_modaddr isn't getting translated

  The trans modules (user/system state address space translation) for MIPS3000, MIPS32, MIPS64, SH-3, and SH-4 no longer fail to translate the input parameter to _os_modaddr(). This was causing modules to be reported as not found when they actually existed.

- CF 12957, 14091: The makdir() and ev_creat() functions do not pass parameters correctly.

  The following functions from sys_clib.l/sys_csl.l were fixed to fill errno properly: alm_atdata(), alm_atjul(), alm_set(), creat(), ev_creat(), and makdir(). In addition, lseek(), ibrk(), ebrk(), getstat(), and setstat().

- CF 13021: The 64-bit integer left shift operator does not work.

  Previously, the front-end failed to promote left shift operators to the required width. This lead the back-end to produce incorrect code. The front-end was fixed to perform left shifts in the proper width.

- CF 13159: The module.h file's subroutine numbers are incorrect.

  The related values and comments related have been corrected. A new module.h file has been created to resolve this issue.

- CF 13291: The mktemp() function does not work correctly.

  The mktemp() function will now correctly set errno when it runs out of filenames to return. This function also ensures that the directory specified in the prefix exists before it returns any filenames. It will now handle any pattern ending in three or more "X's". Also, mktemp() now returns at least 256 unique file names in a given directory.

- CF 13378: The 64-bit "+=" can cause a bad mnemonic to be emitted for PowerPC.

  A typographical error in the PowerPC back-end has been fixed to allow the handling of a 64-bit ident += const construct, where ident is in a register. Programs containing this and other similar constructs can now be compiled.

- CF 13388: The front-end receives a General Protection Fault (GPF) after some inline _asm() statements.

  The front-end received a GPF when it encountered a string constant in a macro invocation after an inline _asm() statement. This problem has been corrected.

- CF 13512: The short-data model on the ARM processor seems to use too-large of a symbol on an add instruction.

  The executive was fixed to ensure that the files are passed to the back-end in the same order they are passed to the linker during data area layout performance. The linker no longer prints errors related to symbol offsets in this situation.
- CF 13564: The wrong prototypes for the `_os_exec()` and `_os_fork()` functions are compiled.

  The prototypes for `_os_exec()` and `_os_fork()` were fixed so they could be successfully compiled in C++ mode.
- CF 13570: `_OPT_PROTOS` is undefined if certain header files are used.

  The following header files no longer define `_OPT_PROTOS`: UNIX/`os9def.h`, UNIX/`os9time.h`, UNIX/`stat.h`, UNIX/`stringlist.h`, and UNIX/`times.h`.
- CF 13591: Using the `signal()` function in C++ delivers an error.

  The `signal()` function is now properly prototyped so it can be compiled in C++ mode.
- CF 13743: The assembler cannot handle illegal expressions.

  Illegal expressions, such as `==label` and `'xxx`, were not handled well by the RISC assemblers. The `==label` expression resulted in no code generation for the line, while `'xxx` resulted in a GPF in the assembler. These problems have been corrected.
- CF 13869: Fix failed support for 64-bit integers on PowerPC.

  Previously, several features of 64-bit integer support failed for PowerPC. These problems have been corrected.
- CF 13978: RISC assembly optimizers fail to support the `lo()`, `hi()`, and `high()` functions in all contexts.

  The RISC assembly optimizers were fixed to allow `lo()`, `hi()`, and `high()` in all valid situations.
- CF 13999: Add PCF support to `unix.l`'s `utimes()`.

  The `utimes()` function in the `unix.l` C library was modified to include support for updating the times for a file on a PCF device (MS-DOS format).
- CF 14074: Arrays of `const` pointers do not work with `const` structures.

- Previously, the code generated to handle `const` qualified pointers within arrays within structures was not working correctly. The front-end now correctly generates code for such data structures.
- CF 14960: PCMCIA utility for ThinClient doesn't work

  The pcmcia would cause user permission errors when it was run (000:164). To remedy the problem it has been recompiled with the group.user option set to 0.0.

Networking Notes

The following sections include the release notes for SoftStax and LAN Communications version 4.1.

Enhancements

The following list describes general enhancements made to SoftStax and Lan Communications for this release. Where applicable, Customer First incidents are included.

- `spf.h`

  `SPF_SS_ROUTEMSG` now adds a route with `rt_msghdr`.

  `spf_pps_idata` is changed to `spf_pps_resv`.
- IP functions have been moved to `spfuncs.h` and the legacy LAN Communications stack's IP functions have been moved to `spfuncs4.h`.
- `spproto`

  The `pullup` function has been added in `dr_downdata()` for hardware drivers that do not handle mbuf packet chain.
- CF 13615: `mbdump` should provide free lists.

  The `mbdump` utility should have an option that prints an mbuf allocation table. The `-a` command line option was added to provide this functionality.
- The `sendto()` function supports item address structure for the legacy LAN Communications stack.

Drivers

- `sp21040`

  IP version 6 (IPv6) multicast support has been added in Edition 36 of the `sp21040` driver. In Edition 73, chain mbuf transmit is supported

without the need for mbuf scooping. The mbuf scooping ability has been removed from the `dr_downdata()` function. In addition, big-endian support has been added for the ARM processor family.

- `spquicc`

  IPv6 multicast support has been added in Edition 28 of the `spquicc` driver.

- `spfa311`

  Multicast for big-endian processors and IPv6 support are available in Edition 5 of this driver. Also, chain mbuf transmit is supported in Edition 6; to select it, set the DP_DESC_CMDSTS_MORE bit.

- `sppro100`

  Multicast for big-endian processors is now supported; swap is removed. In addition, IPv6 multicast support has been added in Edition 19 of this driver. In Edition 20, a compiler warning has been removed. The ability to transmit multiple data buffers using TBD array has been added in Edition 22. Also, chained mbuf support, `SPF_SS_PROMISC` support, and `MEDIA_OPTIONS` support have been added.

- `sp8139`

  This driver now supports IPv6 multicasting and chain mbuf transmitting. Because the chip cannot handle non-contiguous data on transmit, scooping mbuf is used to transmit chain mbuf.

- `spethpq2`

  The `spethpq2` driver now supports IPv6 multicast. Chain mbuf transmit using the feature of the chip has been added and now works on the slow and fast ports. Previously, you could not reuse the first transmit descriptor of a frame on a slow port unless the last descriptor of the frame had been processed by the chip. The driver then stored the Ethernet packet and its device entry in the same cache line. This caused the device entry to become invalidated when the Ethernet packet data was invalidated. This problem has been fixed.

  Other enhancements for the `spethpq2` driver are as follows:

  - Multiple FCCCs are now allowed on the same board.
  - MII interrupts are now available for PHY changes.
  - Support has been added for promiscuous change and restore.
  - Mbufs are no longer consumed if the driver is overloaded.
  - A `qestat` program has been added to report driver statistics.

  - IRQ has been changed to properly clear CCE bits.
- `sp83902_sh`

  The `sp83902_sh` driver now supports IPv6 multicast and chain mbuf transmitting.

Utilities

- `route6d`, `rtsol`, and `ping6` have been added.

  `ftp`, `ifconfig`, `netstat`, `ping6`, `route`, `route6d`, and `rtsol` have been ported from NetBSD version 1.5.1.
- `ifconfig`

  Stop and start network interfaces are now supported.
- `mbdump`

  The `-a` display allocation table option has been added.

Descriptors

- `ip0_router`

  The `spip` descriptor has been added.

NFS

- The following modules have been eliminated: `dird`, `on`, `rcopy`, `rdir`, `rexd`, `rexdc`, `rldd`, `rload`, `rpcgen`, `rpr`, `rsort`, `sorted`, `spray`, and `sprayd`.

Libraries

- `netdb_local.l` and `netdb_local.l` have been eliminated.
- `recvmsg()` and `sendmsg()` socket library functions have been added. In addition, the `flags` parameter for `send()` is now supported.

Protocol Modules

- `spip`

  Multiple rejecting network routes have been added at a start time.

  According to *RFC 1122*, 127.0.0.0/8 should not leave the node 127.0.0.0 `netmask` 0xff000000 127.0.0.1.

Resolved Problems

This section discusses problems that were resolved for SoftStax and LAN Communications. Where applicable, Customer First incidents are included:

Drivers

- `spenet`
  - `deventry` in the mbuf will now determine the incoming interface if the `lu_pathdesc` is set.
  - An Ethernet header has been created to pass hardware addresses to the `spenet` packet.
  - The `ip_rtmsg_pb` parameter block is now used for calling the `IP_SS_RTMSG` setstat.
- `sp91c94`

  In previous editions of this driver, a software bug existed for setting the hash table. This problem has been fixed in Edition 12. Also, multicast support has been added for IPv6.
- `sp860t`

  Previously, this driver did not correctly calculate the hash mask for multicast filtering. This problem has been fixed. The `sp860t` driver now supports IPv6 multicasting and chain mbuf transmitting. The transmit FIFO underrun problem had occurred when the Ethernet port was connected to the 100M switch. This problem is solved when only the last bit is set while on the last mbuf of a pchain.
- CF 12697: Incomplete mbuf search leading to premature `ENOBUFS`

  The TCP/IP stack was fixed so that it no longer prematurely return `ENOBUFS` in rare circumstances related to non-blocking sockets.
- CF 12883: The Intel Ethernet Pro/100 driver hangs upon exit.

  Previously, an `ifconfig stop` command on an interface using the Intel Ethernet Pro/100 caused the system to hang. This problem occurred because the driver did not correctly free the mbufs it had allocated. The driver now frees the correct mbufs.
- CF 12953: Low-level Ethernet drivers need to be enhanced for non-standard networks.

  The suite of low-level Ethernet drivers (those used for system-state debugging and Ethernet booting) were enhanced to better handle non-standard network masks.

- CF 13074, 11110: `sp91c94` Ethernet driver receiver problem.

  The 91c94 memory configuration in the high-level driver (CF 11110) and low-level driver (CF 13074) is now set at initialization time to allow for enough memory for at least one transmit Ethernet packet.
- CF 13232: The 3com PCMCIA Ethernet card does not work on Assabet daughter board.

  The PCMCIA slots on the Assabet daughter board contain different IRQs than the PCMCIA slots on the Assabet main board. This has been resolved; the proper descriptors have been created for the daughter board in conjunction with the 3com PCMCIA Ethernet card. In addition, the Wizard files have been updated to include the proper descriptors automatically.
- CF 13289: Multicast problems in Intel Ethernet Pro driver.

  In the previous version, this driver did not track the multicast address once the number of address entries went beyond those allowed in `MULTICAST_FILTER_LIMIT`. When the number of entries went below the limit, the driver switched from multicast promiscuous mode to multicast filter mode. This resulted in other potential problems, all of which have been corrected.
- CF 13407: SPPRO100 modification in `irq.c`.

  Previously, there was a driver locking problem on PPPC in a rare out-of-memory case: the current buffer was the last buffer allocated in the receive ring. This problem has been fixed; no longer is the last buffer consumed nor the device put into a “no resource” state.
- CF 13560: The `sp860t` driver transmit FIFO is underrun when it connects to the 100M switch.

  Previously, the `sp860t` driver dropped packets; transmit FIFO underruns occurred when the sending packets consisted of mbuf chains. The driver now processes all mbufs in the chain before it transmits.
- CF 13618: The `sppro100` driver hangs when sending multicast addresses.

  A chip cannot resume from suspended state and execute the next setup command if multicast setup frames are processed back to back. The driver has been fixed to alternate between two setup frames instead of one.
- CF 13706: `sp91c94` transmits bad packets if passed an odd mbuf offset.

The sp91c94 does not correctly transmit packets that start at an odd mbuf offset.This problem has been fixed and the driver now handles odd aligned data as well as packet chains.

- CF 13829: MCP750 does not boot when ll21040 is included in the coreboot.

  This problem resulted from a missing value in a switch statement in the ll21040 driver. The additional value has been added so the driver will work on all known MCP750 boards.

- CF 13962: The spquicc driver does not set Mbuf flags correctly.

  In some situations, the spquicc driver sets the Mbuf flags field incorrectly. This problem has been fixed.

- CF 14130: spixeth packet padding problem.

  The spixeth driver does not use the hardware's ability to pad short Ethernet packets when using gigabit ports. The driver has been changed to allow this ability.

- CF 14280: spixeth does not allow multiple paths open.

  Edition 13 of the spixeth driver only allowed one path to be opened on a particular port. This prevented an application from using the SPF_GS_ETHADDR getstat to determine the hardware address. The problem applied when using such applications as DHCP. This was fixed in Edition 14 of the driver.

Protocol Modules (IPv4)

- CF 13235: TCP/IP can consume all Mbuf space.

  If an application was writing to a socket and getting responses, but failing to read the responses, it could cause the TCP/IP stack to consume all the mbuf space in an inefficient manner. The spip module has been fixed to more efficiently store the responses so that they no longer consume an unnecessarily large amount of space.

- CF 13249: TCP does not respond if no ACK bit is set.

  Some broken implementations of TCP do not set the ACK bit when sending TCP packets. The input routine of sptcp has been changed so an ACK is sent in response to packets with no data and no ACK bit set.

- CF 12580: The transfer rate is very slow when the TCP_NODELAY flag is used.

sptcp now better handles the timing consideration when transferring small amounts of data when the TCP_NODELAY option is set.

- CF 12944: Lan Communications memory leak.

  The accept() socket function leaked two lock descriptors each time it was called. This has been fixed; the lock descriptors are now freed when the socket closes.

SPF File Manager

- CF 13139: SPF_PPSTAT_INIT is not filled in the per-path storage structure.

  This problem was the result of inadequate documentation for the functions in ppstat.l. These functions are now documented.

- CF 13262: SPF is not passing an error marked in Mbuf from the driver up stack.

  The SPF file manager was returning SUCCESS in some situations where an error was encountered. SPF was fixed to pay more attention to the error flag on inbound Mbufs.

Utilities

- CF 12612: BREAK and IP options are not handled by telnetdc.

  The telnet daemon (telnetdc) did not handle the BREAK and IP telnet options. These options are now processed and a BREAK option will send a SIGQUIT signal to the child process. The IP option sends a SIGING signal to the child process.

- CF 12789: The DHCP client is confused by bootp responses and floods DISCOVER packets.

  The dhcp utility is no longer confused by bootp responses. It has been fixed to better classify responses.

- CF 12792: The DHCP client "leaks" 32K bytes of Mbuf space.

  This problem has been corrected; dhcp now both sleeps for the IP address lease time and watches for incoming data on the socket. Any incoming data not related to DHCP is discarded, thus freeing the mbuf space.

- CF 13422: telnetd -t option does not auto-logout after a specified time.

The -t option is ignored by telnetd if it is followed by an equal sign (i.e -t=5). This problem has been corrected.

- CF 13522: The DHCP client can override local settings with those returned by server.

  The -override option was added to dhcp allowing existing network configurations, such as DNS name servers, to be overridden by dhcp server-supplied information.

NFS

- CF 11091: NFS server problem.

  The mount utility was fixed to not unlink from a module twice. This caused an exception that was visible on the MIPS platform.

- CF 12629: The NFS server returns an error when a file is opened.

  If an application attempts to open an existing file mounted on the OS-9 NFS server with "write" permission, it receives an 000:203 error from fopen(), even when file permission and UID allow it.

Low-Level ROM

- CF 13450: Ethernet boot memory leak.

  Previously, sockets were not being released for reuse after timeout. llbootp has been modified to release sockets and free memory when no server is found.

- CF 13589: llfttpboot runs longer than normal and writes inappropriate memory.

  lltftpboot no longer writes over extra memory.

- CF 13694: llbootp packets that are sent twice from the server should not be received.

  lltftp no longer ACKs duplicate packets.

IP Infusion

- CF 12538: Command line help does not always work.

  In some situations the command line help using the “?“ button stops working for the IP Infusion routing daemons. This problem existing when an attempt was made to parse invalid commands; it has been resolved.

Known Issues

This section describes the known networking issues that exist for this release. Workarounds are provided, where possible.

General Issues

- Currently, there is no multicast routing protocol (such as DVMRP) that enables multicast forwarding under OS-9.
- IPSec is currently not supported. Multicast routing support is not present in the newly supported LAN Communications stack.
- The newly supported LAN Communications stack does not support tunneling.
- The newly supported LAN Communications stack requires more system stack than the legacy LAN Communications stack. In addition, the legacy stack produces an mbuf chain for transmit and any hardware drivers that do not support the mbuf chain transmission create raw data packets on the wire, and network communication does not establish. Until the source code for such drivers are fixed, the `spproto` driver can be inserted above the hardware drivers to scoop the chained mbuf temporarily.
- Small mbuf pools cause aberrant behavior and occasional bus traps while sending large packets through the `ping` utility.

  If the size of the Mbuf pool is too small to handle a contiguous Mbuf of the size specified, it does not respond to large ping packets.
- `ifconfig unbind`, `stop`, and `down` options do not work while `routed` and/or `route6d` is running.
- OSPF6: The reserved fields in DB description packets should always be zero; however, because structures are not initialized to zero, other values for reserved fields are possible.
- OSPF6: OSPF6D generates an error for the checksum of the LSA header in DB description packets. Though the checksum is correct, OS-9 cannot recognize it because OS-9 does not perform a byteswap before reading the checksum.

SPF File Manager

- When a `read()` on a socket is interrupted by a signal, the `read()` call returns -1 and set errno to `EOS_SIGNAL`.

- If a connection is terminated from the remote side, `_os_read()` now returns an `EOS_EOF` error.
- There is no `recvmsg()` or `sendmsg()` support on the legacy LAN Communications stack.
- `do_read()` returns errors from Mbufs correctly in `IO_PACKET`.

Utilities

- `ftp`

  The mshell utility interprets the "%" sign as the start of a macro. To make this command work, you must quote the URL. This command works if you call it as follows:

  ```
  $ ftp "ftp://emiy@cypher/%2F/tmp/test"
  ```

- `netstat`

  non-contiguous `netmask` is displayed as follows:

  ```
  $ route add -net 208.252.0.0 192.168.3.225 208.252.0.0
  add net 208.252.0.0: gateway 192.168.3.225
  $ netstat -rn
  Routing tables
  Internet:
  Destination        Gateway            Flags     Refs     Use
  Mtu  Interface
  default            192.168.3.225      UGS          0        0
  1500  enet0
  127.0.0.1          127.0.0.1          UH           0        0
  32980  lo0
  192.168.3          192.168.3.103      U            2        0
  1500  enet0
  192.168.3.103      127.0.0.1          UHS          0        0
  32980  lo0
  208.252&0xd0fc0000 192.168.3.225      UGS          0        0
  1500  enet0
  Internet6:
  Destination                   Gateway                   Flags
  Refs     Use    Mtu  Interface
  ::1                       ::1                 UH        0
  0  32980  lo0
  ```

  Note the following line:

  ```
  208.252&0xd0fc0000 192.168.3.225     UGS        0
  0    1500  enet0
  ```

  The line above shows how `netstat` displays a "non-contiguous"

netmask. The value 0xd0fc0000 is the netmask 208.252.0.0 that was specified in the route command above. A netmask is considered
non-contiguous if (while working from bit zero to bit 31) once the first bit that is turned on is found, the rest of them are off.

Attempting this again using a contiguous netmask displays the following content:

```
$ route add -net 208.252.0.0 192.168.3.225 255.255.0.0
add net 208.252.0.0: gateway 192.168.3.225
$ netstat -rn
Routing tables
Internet:
Destination        Gateway            Flags     Refs     Use
Mtu  Interface
default            192.168.3.225      UGS          0        0
1500  enet0
127.0.0.1          127.0.0.1          UH           0        0
32980  lo0
192.168.3          192.168.3.103      U            2        0
1500  enet0
192.168.3.103      127.0.0.1          UHS          0        0
32980  lo0
208.252/16         192.168.3.225      UGS          0        0
1500  enet0
Internet6:
Destination                  Gateway                  Flags
Refs     Use    Mtu  Interface
::1                        ::1                  UH        0
0  32980  lo0
```

OS-9 Utilities Notes

The following section represents changes and updates to OS-9 utilities for this release.

Enhancements

The following list describes the general enhancements made to the OS-9 utility set for the current release. Where applicable, Customer First incidents are included:

- CF 12355: The mshell utility has been enhanced to allow the print() directive to send output to stdout instead of stderr. This enhancement was implemented as a command line option (-w/-nw).

By default, the `print()` directive will write to `stderr`. If `-w` is specified on the command line of in a script file, it will write to `stdout`. This new option is also inherited through the `_SHELLPARAMS` environment variable.

- CF 13814: The mshell utility should have a `%status` variable.

  The mshell utility was enhanced to include support for a shell variable called `%status`. It contains the exit status of the most recent command executed. Refer to the chapter on mshell in the *Utilities Reference* manual for more information.

- CF 13842: `Procs -x` needs to be updated to show newer Process State Flags.

  The following statuses have been included in the list of statuses to be printed when `-x` option is used with `procs`:

  - `FpuProc` for `FPUPROCBT` (process is using the FPU)
  - `RthProc` for `RTPROCBT` (Representative Thread process)
  - `DbgProc` for `DBGPROCBT` (Debug process)

Resolved Problems

This section describes Customer First incidents related to the OS-9 utilities and how they were resolved for the current release.

- CF 9265: PCF does not allow removal of write permissions.

  The case statement for setting the permissions of directories was returning `read/write` each time, regardless of what permissions were set with the `attr` command. This has been modified to return only `read` if the command `attr -nwngwnpw` DIR is issued. The directory's attributes are correctly set from there.

- CF 12852, 13213: `os9make` under Windows gets long command lines wrong.

  Under early versions of Windows/DOS, there is a command line length limit of 128 characters. In order to preserve compatibility with these old versions of Windows/DOS, if there is a command line that exceeds 128 characters, `os9make` would break up the command line and put it into a temporary file, then execute the command with the `-z` option pointing to the temporary file. Although this works with many OS-9 utilities, there are OS-9 utilities that do not support the `-z` option, and no non-OS9 utilities have the option. Because of this, the work-around to break up the command line is no longer valid.

Under newer versions of Windows, the command line length limit was extended; however, this limit varied among versions of Windows, so there was no way of discovering the limitation. This has been resolved;
the `-c` and `-cl` options have been added to `os9make` under Windows. The `-c` option ignores a command line length limit, while the `-cl` option allows a specific command line length to be specified by the user.

- CF 12861: The `os9deldir.exe` file in `MWOS/DOS/BIN` does not work.

  `os9deldir` was added to the cross-hosted utilities. The `-x` option is unavailable, but the `-q` option is always on.

- CF 13078: `mkdatmod -r` is not extracting information from a file.

  `mkdatmod` will extract the information from a module if you specify the size of the original file using the `-s` option.

- CF 13399: `fixmod` required write permission on files it only needs to read

  `fixmod` was fixed to only request read permission when no writing will be required. This is the case when dumping the modules or examining the input file.

- CF 13448: Added an uncompress bootfile option for the `mbc` utility.

  `mbc` Edition 4 or greater has been updated to include an uncompress option (`-u`).

- CF 13684: Redirection is not complete in `mdir`.

  Previously, not all modules were listed by `mdir`. The `_os_exit` calls were modified in `mdir` source code to exit normally; `_os_exit` does not flush the `stdio` buffers. This corrected the problem.

- CF 13764: `os9make` crashes when calling to deeply with sub-makes.

  The array that stores the number of include files when using a for loop was allocated with storage for just 33 files. The size of the include file array for this instance has been increased substantially.

- CF 14134: `mdir -e` does not display the type `cdb` correctly.

  The type for a `cdb` module formerly displayed as "`5`" and is now displayed as "`cdb`".

- CF 14147: `mdir` always show module directories

  `mdir` was fixed to properly exclude module directories from listing that should not include module directories.

- CF 13942: `dir` command causes and exception on long filename

  The `dir` utility was fixed to handle file names up to 256 characters long. The old limit was 44, which was based on RBF's limit for file names.

MAUI Notes

The following sections contain release notes for the OS-9 Compiler.

Resolved Problems

This section gives a description of Customer First (CF) incidents related to MAUI and how they were resolved for the current release.

- CF 14560: `jview` gets Non-Fatal MAUI error

  A MAUI non-fatal error would occur every time the `jview` demo was run. This was caused by a call to an unnecessary call to `gfx_term`. The call to the `gfx_term` function has been removed and the error no longer occurs.

Known Issues

This section describes the known MAUI issues that exist for this release. Workarounds are provided, where possible.

- CF 14562: bold25.fnt module is not 4 byte aligned.

  This causes errors such as 103:008 for ARM and SH processors. It is recommended that a different font module is used.

- CF 14581: winink maui demo problems

  The winink demo has problems coinciding its window area with its drawing area. This problem manifests itself in various ways such as being able to ink onto the window manager or other windows.

Add-On Notes

The following section contains release notes for OS-9 Add-On Components.

Resolved Problems

This section gives a description of Customer First (CF) incidents related to Add-Ons and how they were resolved for the current release.

- CF 14584: Tech Check Gets memory access error in certain circumstance

  In certain cases TechCheck would get memory access errors when trying to save an incident report as text. This problem has been fixed in version 1.1.3.2 of tcheck.exe.

- CF 14706: error in SNMP vcr agent makefile

  Fixed Makeifle.os9 in /mwos/SRC/SPF/SNMPLITE/src/snmp/extend/examples/vcr/byhand so it will no longer try to build snmpd.

- CF 14717: Remove UK and French support offices from Tech-Check

  The UK and French support offices were removed from TechCheck in version 1.1.3.1.

- CF 14738: NetFront build error

  Due to changes in netdb.h, the NetFront file NETFRONT\SOURCE\AVEFRONT\WAVE\OS9\peer.c needs to move the following block of code above the line that states "`#include <netdb.h>`".

  ```
  #define _OPT_PROTOS
  #include <modes.h>
  #include <UNIX/os9def.h>
  #undef _OPT_PROTOS
  ```

  This will fix the following errors that would otherwise arise while building the NetFront project in Hawk.

- CF 14862: Applet support using NetFront browser causes NumberFormatException

  This problem was resolved by changing the format of the MAUI window ID passed from native code to Java code (changed from unsigned to signed to accommodate ARM's 0xcXXXXXXX addresses).

Known Issues

This section describes the known issues for the Add-Ons that exist for this release. Workarounds are provided, where possible.

- CF 14694: USB peripheral doesn't work if cache is on the board.

  The hawkview utility doesn't connect via USB if cache is enabled on the board. This is a timing problem with the spusbdsl11 driver.

A LAN Communications Stack Migration Reference

This appendix exists as an informational tool for customers migrating from the previously supported networking stack (that which supports the `AF_INET` address family) to the newly supported networking stack (that which supports both `AF_INET` and `AF_INET6` address families).

It is recommended that you begin implementing the new stack with this release of OS-9; information regarding the previously supported stack will not be documented in subsequent releases.

Definitions

In this appendix, the following terms are used:

- "Legacy" refers to the LAN Communications stack that supports the `AF_INET` address family; this is the only network stack supported prior to OS-9 version 4.0.
- "Current" refers to the newly implemented LAN Communications stack, which supports both `AF_INET` and `AF_INET6` address families. This stack is supported for OS-9 version 4.0 or later.

Migration from Legacy to Current Stack

This section discusses how the current stack has been implemented for OS-9. By default, utilities, tools, and libraries include `AF_INET6` support, where possible.

Utility Updates

Utilities for the current stack are ported with the NetBSD and OS-9 functionality in mind. The following list details how certain utilities have changed with implementation of the current stack.

- `telnet`, `telnetd`, and `telnetdc` are still based on the legacy OS-9 code, but have been enhanced to support IPv6 addresses. In addition, they each use the same command line parameters as the `AF_INET` family versions.
- `ftp`, `ftpd`, and `ftpdc` have been ported from NetBSD to OS-9 and, as such, have different command line parameters and require different system configuration. The NetBSD port includes additional functionality, including PASV support.
- `ping` is supported by the `AF_INET` family only. A new utility, `ping6`, has been created for `AF_INET6` support. `ping6` is currently ported from NetBSD sources.

The Code Base

The current stack is based on NetBSD 1.5.1 code. NetBSD IPv6 was first officially supported in version 1.5, based on the KAME Project NetBSD enhancements. The KAME Project is a joint effort to create single, solid software set, especially targeted at IPv6/Ipsec. The IPv6 code from this KAME Project was merged into NetBSD in June 1999. It is included in the NetBSD 1.5 official release.

Updates to Network Configuration Files

The current and legacy stacks contain the same IP address configuration. In addition, the configuration files live in the same directory and work for both the legacy and current LAN Communications stacks. However, when you add new IP addresses for the `AF_INET6` family, it is required that you use IPv6 keywords and addresses for the configuration files.

The network configuration is defined in a set of text configuration files that are compiled into a set of data modules that are consumed when the network stack is started. The `AF_INET` family functionality works between the legacy and current stacks.

In practice, host configuration activities are performed from within the OS-9 Configuration Wizard, which has been enhanced to support multiple Ethernet interfaces and IPv6 addressing requirements.

Router configuration is currently unsupported; thus, manual configuration is required.

Ethernet Drivers

Existing Ethernet drivers work under the current stack without any modifications. However, IPv6 Neighbor Discovery requires multicast support; if you intend on using the current stack, be sure that the drivers you are using support this feature.

The current stack uses mbuf packet chains, while the legacy stack avoided packet chains all together. The packet chain needs to be transmitted in a single Ethernet packet.

Do not loop on the packet chain to transmit separately. This will yield raw data without an IP header going out on the wire.

If the hardware does not support transmitting from multiple buffers, an enhancement to the driver to combine the mbuf packet chain to a single mbuf is recommended. If you do not have hardware driver source code to correct the mbuf packet chain issue, simply place the `spproto` driver above the hardware driver to gather an mbuf chain to a single Ethernet packet.

Miscellaneous Updates

- `netdb_local` has been eliminated. It is possible, however, to disable DNS lookup via other configuration methods.
- The legacy stack's headers and utilities have moved to subdirectories called `BSD4`. The `lancom.tpl` makefile template includes a switch to select the old or new headers. By default, compilations will use the current stack's headers and utilities at the directory locations. To use the legacy stack's headers and utilities, insert `LANCOM = 1` into the makefile definition.
- In general, legacy stack application binaries that are linked with the `netdb` trap library (`netdb.l`) should work with the current network stack; recompilation is not necessary. However, the current network stack may require additional stack space, which can be specified on the command line (e.g. `myftp foo #10`) or set permanently in the module (e.g. `fixmod -us=10 myftp`).