# OS-9 for MIPS Release Notes

## Version 3.1

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 D
November 2001

## Copyright and publication information

This manual reflects version 3.1 of Enhanced OS-9 for MIPS.

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.

# Table of Contents

# Chapter 1: Introduction

As part of Microware's policy of continued product development, OS-9 for MIPS version 3.1 represents a maintenance and update release to incorporate all of the improvements that have been introduced into the component parts.

In addition, MIPS32 is the only processor supported by RadiSys Microware Communications Software Division, Inc. that is released in the version 3.1 configuration. The next version release for other processors will be Enhanced OS-9 version 3.2.

Two of the more significant upgrades for this release are listed below:

- **Board Level support**. This release introduces board level support for the following MIPS32 processors:

  MIPS32 334: This board uses an IDT communications processor to provide full PCI support.

  MIPS32 355: This board provides full on-chip networking support.

  MIPS 4KC: A MIPS 4KC processor daughter card has been implemented to run on an Atlas board, which demonstrates the MIPS32 specification.

- **Hawkeye and DAVID**. Hawkeye and DAVID are now available for the MIPS32 processor family.

## Note

These release notes cover the changes made to the Enhanced OS-9 for MIPS32 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 for MIPS32 version 3.1 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: Operating System

This chapter provides an overview of the changes and improvements made to OS-9 since version 3.0. The following sections are included:

- **Operating System Notes**

# Operating System Notes

The following sections represent changes and updates for OS-9 version 3.1. The OS-9 kernel edition is 1.44.

## Header Files

- The prototype for tputs() has been changed in /MWOS/OS9000/SRC/DEFS/termcap.h. Previously, the prototype was:

  ```
  void tputs(char *, int, int (*)())
  ```

  It has been changed to:

  ```
  void tputs(char *, int, int (*)(char))
  ```

  The function that is passed as the third parameter to tputs() is passed an argument of type char.
- The header file /mwos/src/defs/signal.h has been changed to define fatalsignal() and SIGDEADLY.

## Kernel and System Modules

The following information represents changes made to the kernel and system modules.

- The kernel has been changed to correctly handle unlinking module groups. Previously, the kernel could return the EOS_BPADDR error when unlink a group of modules.
- A problem was fixed in the sc16450 driver. Previously, the driver could have caused the port to quit working.

## Libraries

- A problem has been fixed in the `lock.l` library. The problem could have caused the system to crash if applications created and destroyed a large number of links. Applications that held more than 64 locks, released the locks, then held more then 64 locks again, could have caused a system crash.
- In order to use the `clock()` function from a system state module, the global variable `_environ` must be defined. The following C declarations can be used to properly define and initialize `_environ`:

  ```
  void *xenviron;
  ```

  ```
  void **_environ = &xenviron;
  ```
- When using `_os_ioconfig()` to increase the number of paths, an I/O error or exception would occasionally occur. This problem has been fixed.

## Makefiles

- Previously, the `MWOS/SRC/DPIO/SPF[MFM]/DESC/desc.tpl` files used `os9merge` together with the “>” redirection to create a new file. This caused a problem under the PVCS shell where the > automatically converts 0x0a to 0x0d0a, which adversely affects the output. The template files have been changed to use the `os9merge -o` options in stead of the > symbol for this.

## Rombug

- A new command, “`s.`”, has been added to Rombug. The command performs the same function as the “s” command (list symbols), but does not perform a page pause when listing the symbols.

## System Modules

- Under certain circumstances, `ioman` tried to return memory of length `0` during an attach. This has been fixed.

## Low Level/ROM Code

- The low level driver for the Intel Pro100 (`llpro100`) will now work correctly in polled mode if on a busy network. The driver will now set up the receive rings to suspend the Intel pro100 chip correctly when the rings have been filled.

  Previously, because driver was in polled mode, no IRQ occurred to keep the receive rings drained. On busy networks, the receive rings would rapidly fill up with broadcasts. This was caused by the fact that the Intel pro100 chip was not suspended correctly; therefore, the whole chip and driver would crash.

- The bootp booter service has been enhanced to keep the response from the bootp server so that it is available to system and application code. The following is an example of how to access the server response and dump the information from it.

```
#include <ROM/rom.h>
#include "bsddefs.h"
#include <netdb.h>
#include <netdblib.h>
#include <string.h>
#include <net/if_dl.h>
#include <netinet/in_var.h>
#include <sys/ioctl.h>
#include <stdlib.h>
#include <protocols/bootp.h>
#include <stdio.h>
/********************/
```

```
#ifdef _OS9000 /* Should be included only for Modular ROM based
systems */
#include <ROM/romerrno.h>
#include <ROM/p2lib.h>
#endif

extern struct ifnet loif;
extern struct ifaddr **ifnet_addrs;
void main(void)
{
    struct bootp *bpmsg;
    int *i;
    int len;

    u_char *savedGlobals;        /*  Saved Globals Pointer */
    u_char *sysGlobals;          /* System Globals Pointer */
    Rominfo       rinf_p;       /* pointer to rom info structure */
    Proto_srvr      psrvr_ptr; /* pointer to walk protoman list */
     printf("hello\n");

     /* Get System Globals Pointer */
     sysGlobals = (u_char*)get_sysglobs();

    /* Save current Globals Pointer */
    savedGlobals = change_static(sysGlobals);
    /* Now that static pointer points to system Globals..call
   getrinf */
    if (( getrinf(&rinf_p)) != 0) {
        /* restore old globals pointer before exiting */
        change_static(savedGlobals);


    } else {
        /* restore old globals pointer before continuing */
        change_static(savedGlobals);
           printf("hi\n");
        /* check the version of the Rominfo structure *
        /*
         * Struct is good .. get pointer to protocol list
         * Traverse list looking for valid protocol Driver structs
         * finding ones that have the BOOTDEV flag set
        */
         psrvr_ptr = (Proto_srvr)rinf_p->protoman;
        while (psrvr_ptr) {  printf("good\n");
        /* printf("the psrvr_ptr is %d\n",psrvr_ptr->struct_id);
               printf("PRTSVRID VALUE IS %d\n",PRTSVRID);
               printf("value of struct_ver is
%d\n",psrvr_ptr->struct_ver);
               printf("value of max version is %d\n",PVR_VER_MAX);
               printf("the value of proto_flag
%d\n",psrvr_ptr->proto_flags);
               printf("the value of bootdev %d\n",PVR_BOOTDEV);
               printf("the value of driver %d\n",PVR_DRIVER);*/
```

```
              if ((psrvr_ptr->struct_id == PRTSVRID) &&
                /* NOTE: UPPER Bounds check only ... add lower if
needed */
                (psrvr_ptr->struct_ver <= PVR_VER_MAX) &&
                (psrvr_ptr->proto_flags & (PVR_BOOTDEV |
PVR_DRIVER)) )
              {
                  printf("better\n");
                  bpmsg =(struct bootp *)(psrvr_ptr->proto_bootp);
                     i =(int *)bpmsg;
                     i=i-1;
                     printf("%d\n",*i);

                     printf("%x\n",bpmsg->bp_siaddr.s_addr);
                     for(len=0;len < 64; len++)
                       printf("%x\n",bpmsg->bp_vend[len]);

                     printf("best\n");
             }
          psrvr_ptr = psrvr_ptr->next;
        }
      printf("complete\n");
    }

}
```

# Chapter 3: Host Applications

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

- **Configuration Wizard Notes**
- **Hawk Notes**
- **Hawkeye Notes**
- **TECH-CHECK Notes**

# Configuration Wizard Notes

The following sections represent changes and updates for the Configuration Wizard, editions 1.44 through 1.72.

## Enhancements

- CF7029: Configuration Wizard enhancement requests:

  1. In the **Build Image** window, if you click **Build**, the Wizard displays the following message: “Building Creating BootFile Image...seconds XX.”

     The text has been changed to, "Creating Bootfile Image...seconds XX.” Also, "BootFile" has been changed to "Bootfile".

  2. The Wizard is not friendly to someone who attempts to use it on an `MWOS` tree inside a revision control system.

     An enhancement has been made to the Wizard to correct this problem. Now, when the Wizard is started, a check is done to ensure that `os9p.ini` is writeable. If it is not writeable, an error message will then be displayed telling the user that the file is read-only. If the file is read-only, the user will be required to check the file out of his/her revision control system to make the file writeable.

  3. It would be helpful if the Wizard had tool tips to explain what the various selection boxes are used for on the screen.

     Tool tips have been added.

  4. If the user clicks **OK** without first selecting a configuration name, a prompt appears stating, "Configuration name required."

     The prompt title has been changed and an explanation of the configuration name has been added.

5. The choice **Use Wizard** is too far from the selection point. The choice **Advanced Mode** is too close.

   The choices have been changed to **Beginner Mode** and **Expert Mode** for all platforms. In addition, on the X86 system, **Beginner Mode** is used for system preparation as well. All IDE disk descriptors are included in this mode on the X86. For the X86 even "experts" will use this mode when preparing new systems.

6. In the **Wizard mode**, the **Define Debugger** window does not allow the user to go back. Clicking **Cancel** exits the Wizard.

   The Wizard (Beginner mode) has been modified so that the **Back** button on the first property sheet page is disabled every time it is viewed. The main screen button has been changed from **Cancel** to **Exit**. When the **Cancel** button is clicked in **Beginner mode** the Wizard returns to the initial screen.

7. Every item in the Wizard should have a tool tip explaining in detail what capabilities the selection provides. For instance, what is the "Slip" selection used for? Why is Slip not capitalized?

   Tool tips have been added to the selections. "Slip" was changed to "SLIP".

8. The **Back** button is not available inside the **Define Debugger** screen. After clicking **Next** in the **Define Debugger** screen and clicking back, the **Back** button may be clicked. It still does not return you to the initial screen.

   The Wizard has been changed so that the **Back** button is disabled when the first property sheet page is viewed.

**Note**

The **Cancel** button will return to the initial screen.

9. The Slip Setup for Remote Debugging window has an inside window, which does not provide enough room to display all of the configuration text (for Subnet Mask). Also, it does not ask for an IP gateway. The input fields are also not wide enough to handle a full ###.###.###.### address.

   The input fields were adjusted to handle a full ###.###.###.### address. SLIP does not use an IP gateway because communication is done directly over the serial cable.

10. The **Define ROM Ports** window has text descriptions that have been cut off because not enough room was allocated for their length.

    The length allocated for the text description was changed.

11. In the **Define Other Boot Options** window, what does “OverRide Booter” mean and why is the "R" capitalized?

    "OverRide Booter" has been changed to "Override Booter" and a description has been added.

12. In the **IDE Configuration** window, why is it not possible to add booting both off "Socket #0" and "Socket #1"?

    The "Select Default Boot Device" string was added to clarify this window.

13. The **SPF Setup** screen has NFS information input boxes. The "Startup Directory" text is clipped to the point the user cannot read "Directory.”

    This problem was resolved.

14. The **Init Options** screen allows the user to change the tick rate. In what quantum is this specified?

    The **Tick Rate** field description has been changed to **Tick Rate (Ticks/Sec)**.

15. The **Init Options** has a **Ticks Per Time Slice** window. It may not be apparent to the user that this is for Round Robin task switching.

    The **Ticks Per Time Slice** field description was changed to **Ticks Per Time Slice (Round Robin Task Switching)**.

16. The **Init Options** has a **Wipe Memory Flag** check box. By the nature of a check box, it is already a flag.

    The **Wipe Memory Flag** check box has been changed to **Wipe Memory When Allocated**.

17. The **Build Image** window spells the name of the system debugger as "ROMBug"; the **Define Debugger** spells it as "RomBug". Which is correct?

    "ROMBug" was changed to "RomBug".

18. The **Build Image** screen has an **Include** column. Everything in this column is a set of modules.

    The string "Modules" was removed from all check boxes in the **Include** column.

19. In the **Build Image** window, after the user clicks **Check** or **Build**, there is too much information in the status area (or the present information is not separated cleanly enough).

    The success/failure and the total size (in bytes) of a build is stored in the first box after a build.

20. In the **Build Image** window, if the user clicks **Build**, the Wizard builds an image as expected and the **Save As** box is properly ungrayed. If the user clicks **Save As** and then either clicks **Cancel** or **Save the file**, it returns the window with the **Save As** option grayed out. It cannot be selected again until **Build** is clicked again.

    The Wizard has been changed so that the **Save As** box stays enabled even after a user has saved it or canceled the save. In addition, the user may change the **Build Type/Options** radio button and return to the image that was last built to save it again.

21. In the **Build Image** window, if the user clicks **Save As**, the Wizard allows him/her to save in any chosen directory. If the user elects to save in the same `PORTBOOT` directory as the `os9kboot` file was created, it fails to do this and does not indicate it has failed. The output file is not there, however.

    The Wizard has been modified to allow the user to save the file wherever he/she wants.

22. In the **Advanced mode** of the Wizard, the **About** box is titled, “About OS9P”, where it should be “About OS-9 Configuration Wizard” or something similar. Also, the text in the box says, "OS-9 Port Utility," where it should say, "OS-9 Configuration Wizard" or something similar.

    The text has been changed according to the description above.

23. In the **Advanced mode** of the Wizard, the Configure menu has a Bootfile menu with the words, "NetWork Configuration."

    The "NetWork Configuration" string was changed to "Network Configuration".

24. In the **Advanced mode** of the Wizard, various top level menus display text at the bottom indicating what kind of things each menu can do. The **Configure** menu has text that says, "Get help." The **Sources** menu has no text. The **Help** menu has no text.

    The items in the idPopup array that pertain to the menus have been commented out. This prevents incorrect text from being displayed in the status bar when a popup menu is selected.

25. In the **Advanced mode** of the Wizard, the **Configure** menu has a **Sys** menu with one item, "Select System Type." It does not make sense to have a secondary menu for a single item. Also, this menu item does not appear to do anything. A window appears and disappears very quickly, which is easy to miss. Nothing else changes on the screen and there is no status information.

    The **Sys** popup menu was removed and "Select System Type" is now a menu item in the **Configure** menu.

26. When the build is complete, it would be very informative to the user if the Wizard indicates where the boot file may be found.

    A new text box was added on the **Build Image** window that displays the path of an image after it is built.

- CF7476: Enhancements needed in the Wizard involving the use of hostname(s).

  The Wizard has been changed so that the information in the `<MWOS>\SRC\ETC\hosts` file is appended to the host name specified in the Wizard DNS configuration. These hosts entries are all included in the hosts file that is generated by the Wizard. This allows users to create multiple host entries.

**Note**

The `Build_SPF_HOSTS` function has been changed so that the information in the `<MWOS>\SRC\ETC\hosts` file is included in the hosts file that is generated by the Wizard. This allows users to create multiple host entries.

- CF7839: `mbinstall`; `ipstart` should not automatically be inserted into Wizard init.

  The check boxes **Start mbinstall**, (`IDC_SPF_MBINSTALL`) and **Start ipstart** (`IDC_SPF_IPSTART`) have been added to the **SoftStax Setup** tab under **System Network Configuration**.

- CF10315: Add an `/r0` Initial Device Name radio button.

  This has been added in the **Initial Device Name on Disk Configuration** tab.

- CF11694: Do not destroy dialog until finished with `.ini` files.

  The exit behavior of the Wizard is now changed when a configuration has changed. After a user chooses to save the configuration, the main application window stays on the screen until it has finished writing the configuration information to the `.ini` file.

- CF11696: Extend drop down list.

  The Port Selection and Configuration Name drop down lists have been extended.

- CF11703: Do not show results until finished.

  Changed the Wizard so the final results are not displayed until all of the work is completed.

  If the user has selected the bootp option in the Wizard, (**Configure** -> **Coreboot** -> **Main Configuration** -> **Ethernet** tab: **Add to Boot Menu** and/or **Auto Boot** check boxes), the following calculation is performed and displayed after a build has completed:

  ```
  (boot image size)/512 + 1
  ```

  If an S-record is created, the following calculation is performed and displayed:

  ```
  (S-record size)/512 + 1
  ```

  Four of the five static text boxes on the **Master Builder** window have been replaced with an edit box. This change provides an easy method of displaying additional check and build information, and more text can be printed on each line. Furthermore, the user can now copy and paste the check and build information from the Wizard.

## Resolved Problems

- CF2585: Question mark in upper left corner is not fully functional.

  The `cnfg.c` file has been changed to include a help button in each window when Wizard is in **Beginner mode**. In addition, a help button has been added to the **Beginner mode** to direct users to context-sensitive help.

- CF4026: Wizard mode **Back** button is incorrectly highlighted.

  The Wizard (**Beginner mode**) has been modified so that the **Back** button on the first property sheet page is disabled every time it is viewed. The main screen button has been changed from **Cancel** to **Exit**. When the **Cancel** button is clicked in **Beginner mode**, the Wizard returns to the initial screen.

- CF5543: The Wizard does not complain about unwriteable files.

  The Wizard now searches for the `os9p.ini` file and checks its read-only status. If the file is not present or is read-only, a message is displayed to the user and the program is ended. If the file is not read-only, the program continues.

- CF7038: Wizard Check aborts with error until build runs the first time.

  `inetdb` is the module, which is being built by the Wizard. Therefore, the user is expected to run **Build** prior to **Check**.

### Note

A problem was fixed in which the Wizard Check aborted with an error stating an inability to open `inetdb` or `inetdb2`. This was done by providing dummy file size values for `inetdb` and `inetdb2`, if they had not been built. A message will be displayed to the user about the dummy file size and that he/she can fix the problem by running **Build** and then **Check** again.

- CF7047: The Wizard fails with unknown error message when the `error_log.txt` is write protected.

  The read-only status of the `error_log.txt` file is checked before it is displayed. If the file is read-only, a message is displayed to the user.

- CF7439: `idbgen` error when building with Configuration Wizard.

  The Wizard has been modified so that when a build is performed, host and domain information are required, if DNS is enabled. This fixes the Wizard's problem with `idbgen`.

**Note**

The `BuildDialogProc` function was changed so that when a build is performed, the Host and Domain information are required if DNS is enabled. The `NetDnsDialogProc` function was changed so that when a user selects the **Disable DNS** radio button, the text in the Host and Domain fields are set to their default values if they are blank.

- CF7473: Is there a way to select another`.ini` file from inside the Wizard?

  This change has been incorporated. To select another `.ini` file from inside the Wizard, select **File** -> **New** -> **NewChange_Configuration** from the Wizard menu.

- CF7483: **Parameter List** text box does not stay enabled when user is selected.

  Version 1.45 of the Wizard contains the fix to this problem.

- CF7764: Wizard generates corrupt `cnfgdata` module.

  The Configuration Wizard `<board>.ini` files for the 821 family have been updated to include `-dc_all` in between `-dDATA` and `-dCNFGDATA`. This problem is unique to these boards because only five out of six OS-9 configuration sections are included in the `cnfgdata` module. Adding the `-dc_all` parameter in the `<board>.ini` file ensures that the missing section is included when a build is performed.

  When `EditMod` is run in edit mode on the `cnfgdata` module in a coreboot file with the `-dc_all` parameter (`editmod -e -dc_all cnfgdata -f=coreboot`), all six sections are assumed to be included. If a section is missing, the assumed number of sections (six) will not coincide with the number of sections that are actually present (five). This will result in obscure data in some of the fields. In other words, using `-dc_all` to edit makes `EditMod` think there are six sections instead of five; thus, some of the data presented "appears" corrupt.

- CF8104: Wizard GUI minor problems:

  1. The screen does not refresh when moved behind a window that is "always on top."

     The initial DialogBox is refreshed whenever it is covered by another window and when it is moved off the screen.

  2. Where you have the initial Wizard window open, there is no process icon in the task bar like the other open windows. The Wizard does not appear in the task bar until you select a processor. If you have many windows open on your workstation, the Wizard can get "lost" behind other windows.

     The Wizard has been changed so that the main application window appears before the initial DialogBox. This creates a process icon in the task bar when the Wizard begins.

  3. When you select **OK** from the initial screen, there may be a long delay while it reads the `.ini` files. There is no indication that the Wizard is doing anything.

     The cursor has changed to an hour glass when the Wizard is busy. `MainDialogProc` has been changed so that when **OK** is clicked from the initial screen, the cursor is switched to an hourglass until the program is ready to accept input again. Code has been added in the `MainDialogProc` function to refresh the initial DialogBox when the `WM_SYNCPAINT` and `WM_WINDOWPOSCHANGED` messages are received.

     `SaveIniCnfg` has been changed so that the **Cancel** button is shown in the **Expert mode** when the user is asked if he/she would like to save his/her configuration. If the **Cancel** button is clicked, -1 is returned. The **Cancel** button is not enabled in the **Beginner mode** when a user is asked if he/she would like to save his/her configuration. All buttons have been changed to be initially disabled.

4. The Wizard gets "lost" after a Save As to a different drive.

   This issue has been resolved through a saving of the current directory into a temporary variable prior to doing the SaveAs. This, in turn, generated another error that would not allow the Wizard to save the image to another drive letter. This second problem was corrected by using the temporary variable containing the current directory to generate the source filename used in the CopyFile function.

   A save of the current directory prior to SaveAs of coreboot, bootfile, and rom images has been added. The inability to save to a different drive letter when `MWOS` location is `\MWOS` has been corrected.

- CF9302: Incorrect PCMCIA IDE descriptor naming convention is displayed.

  This is found in the **Configure** -> **Bootfile** -> **Disk Configuration**, **IDE Configuration** tab, PCMCIA IDE section. The naming convention listed for the descriptor in the grayed out pull down menu is incorrect (`/hc1` or `/he1`) until the **PC File System** check box is checked. It appears that the standard IDE descriptors are being listed instead of the PCMCIA IDE ones. Once this check box has been toggled, however, the naming convention listed is correct (unchecked: `pchc1` or `pche1`; checked: `pcmhc1` or `pcmhe1`). Although the incorrect descriptor name may be listed, it appears the appropriate descriptor is still properly included in the bootfile.

  `cnfg.c` has been changed to use correct naming convention upon startup.

- CF9543: Wizard.INI OPTIONS_BOOTFILE bug.

  In the Wizard's.INI file the OPTIONS_BOOTFILE OPTIONS must start at OPTION1, even though skipping options numbers is allowed after that. For instance this fails:

  ```
  ; Bootfile additions

  [OPTIONS_BOOTFILE]
  OPTION2=abort
  OPTION_STATE_2=TRUE
  OPTION21=Thread Support
  OPTION_STATE_21=TRUE
  ```

  while this works:

  ```
  ; Bootfile additions

  [OPTIONS_BOOTFILE]
  OPTION1=abort
  OPTION_STATE_1=TRUE
  OPTION21=Thread Support
  OPTION_STATE_21=TRUE
  ```

  The Wizard should be fixed to allow the selection of any options and not require OPTION1.

  Wizard code was modified to accept an undefined OPTION1 for options that are user definable (spf options, nfs options, utility options, user options, pcmcia options, java options, maui options, coreboot options, and bootfile options).

- CF10196: Parameter list updates are not automatic.

  The Wizard's **Parameter List** field (**Configure** -> **Bootfile** -> **Disk Configuration**: **Init Options** tab) is not automatically updated based on what options the customer has selected. Even if a RAM disk is not enabled, **No Disk Initial Device Name** must be manually selected for the SoftStax relevant strings to be added to the Parameter List.

  This problem occurs when the **User** radio button is selected. This radio button allows the user to define their own parameter list. Because of this, altering their changes based upon the status of check boxes is not recommended.

- CF10256: (Hawk v2.1) **Create a new project** produces an access violation.

  When creating the second project in a project space, an access violation would occur. This problem has been fixed in the current release of Hawk.

- CF10316: `destaddr` versus broadcast address for PPP & SLIP configuration.

  The Wizard **Interface** tab has been changed from "IP Broadcast" to “IP Destination” and from “IP Address” to “IP Source Address”; select the **Subnet Mask** edit box when PPP or slip is selected. `interfaces.conf` file is unchanged.

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

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

- CF10894: **Save As** does not work in the coreboot or bootfile to the `PORTBOOT` directory.

  This problem has been fixed.

- CF10895: DHCP Configuration. (`idbgen` does not like NULL strings.) The Wizard does not include the `dhcp` module. DHCP configuration does not work.

  The Configuration Wizard v1.7 fixes this issue. Although the assign server option was resent, on X86 and other platforms that do not use the PPP/SLIP dialogs there was a bug introduced that did not allow `DHCP_ENABLE` to be set.

**Note**

X86 v3.2 now includes PPP/SLIP dialogs. Only customer specific`.ini` files that do not include PPP/SLIP dialogs require this fix.

The result is the IP information was used as if the DHCP option was not selected and the DHCP module would not be included. Using the new Wizard v1.7 will solve this problem.

Work Around: Although a new version of the Wizard will fix the issue users who want to use DHCP may do so without an update. The following steps detail how to do this:

1. Open the following file:
   `OS9000/80386/PORTS/PCAT/BOOTS/INSTALL/PORTBOOT/spf.ml`
2. Find the `* [DHCP]` line. Change `[DHCP]` to `[]`. The bracketed names are keys. Changing the key to `[]` will force the DHCP module to be included.
3. When setting up IP, select **Specify an IP address**. Make sure the address is set to "0.0.0.0" and that the broadcast and netmask make sense for the network used.
4. Select **Server assigned IP address**.
5. Build boot image with SPF networking support. DHCP should work fine in this configuration and has been tested.

## Note

If DNS is selected, the `resolv.conf` file information is used. otherwise, it is not. The default domain is `alpha.com`, but removing the domain line sill allows building inetdb files.

Tests with DHCP were done with ISC DHCP server for Linux. Many tests were done with the released version of DHCP server for Linux, pre-packaged with RedHat 7.1 SeaWolf. Additional testing was done with DHCP version 3, Release Candidate 12 from ISC.

Below is the sample configuration file used with the ISC DHCP server in conjunction with the tests performed:

```
http://www.isc.org
Version: dhcp Version 3, Release Candidate 12
File: dhcp-3.0rc12.tar.gz
command:
dhcpd -cf /etc/dhcpd/dhcpd.conf -lf /etc/dhcpd/dhcpd.leases
eth0


# /etc/dhcpd/dhcpd.conf


ddns-update-style ad-hoc;# required if using V3 of dhcp
allow unknown-clients;
default-lease-time 1800;# 30 minutes
max-lease-time 7200;# 2 hours


subnet 10.0.0.0 netmask 255.255.255.0 {
    range 10.0.0.20 10.0.0.120;
}
```

- CF11025: Gateway on the Wizard will not remove entry.

  `cnfg.c` has been changed to allow all DNS gateways to be deleted, if desired.

**Note**

`NetGateDialogProc` has been changed to update `rgSetupInfo[0].ether_ip_gateway` when `rgSetupInfo[0].dns_gate` is empty.

- CF11120: The Wizard can generate a bad init string when using NFS.

  The NFS init string has been changed to not include `chx` or `chd` commands unless specified by user.

- CF11188: The Wizard includes the iniz utility multiple times.

  The Wizard previously included iniz in the files listed below. The include for iniz has been commented in each of the `util.ml` files.

  ```
  OS9000/603/PORTS/MOTRAVEN/BOOTS/INSTALL/PORTBOOT/bootfile.ml:../../../../../../PPC/CMDS/iniz
  ```

  ```
  OS9000/603/PORTS/MOTRAVEN/BOOTS/INSTALL/PORTBOOT/util.ml:../../../../../../PPC/CMDS/iniz
  ```

  ```
  OS9000/603/PORTS/MVME1600/BOOTS/INSTALL/PORTBOOT/bootfile.ml:../../../../../../PPC/CMDS/iniz
  ```

  ```
  OS9000/603/PORTS/MVME1600/BOOTS/INSTALL/PORTBOOT/util.ml:../../../../../../PPC/CMDS/iniz
  ```

  ```
  OS9000/ARMV4/PORTS/GRAPHICSMASTER/BOOTS/INSTALL/PORTBOOT/bootfile.ml:../../../../../../ARMV4/CMDS/iniz
  ```

  ```
  OS9000/ARMV4/PORTS/GRAPHICSMASTER/BOOTS/INSTALL/PORTBOOT/util.ml:../../../../../../ARMV4/CMDS/iniz
  ```

- CF11724: The Wizard steals focus while building.

  The SW_MINIMIZE has been changed to SW_HIDE in build.c. When a build was performed the Wizard would steal the focus from the process that currently had focus. SW_HIDE resolves this problem because a DOS window is no longer shown when a process is created.

- CF11725: The Wizard's base window does not retain resize information.

  The Wizard's main application window size information is now retained in os9p.ini.

  ```
  ; os9p.ini

  [DEFAULT]

  RES_WIDTH=1600

  RES_HEIGHT=1200

  WIDTH=438

  HEIGHT=111
  ```

- CF11762: The Wizard is setting the wrong flag if **Wipe Memory Flag** is selected.

  If **Wipe Memory Flag** is selected in the Wizard (on the init-module page), the #define CPUCOMPAT_OVERRIDE B_WIPEMEM is set in the generated config.h. However, the correct way is: #define COMPAT_OVERRIDE B_WIPEMEM. The **Wipe Memory Flag** must be set in m_compat and not in m_cpucompat.

  #define CPUCOMPAT_OVERRIDE B_WIPEMEM changed to #define COMPAT_OVERRIDE B_WIPEMEM.

  Capability added to set advanced compatibility flags (B_GHOST, B_NOCLOCK, B_EXPTBL, B_UDATMOD see <MWOS>\OS9000\SRC\DEFS\init.h) by adding an entry for ADV_COMPAT_FLAGS in the INIT_MODULE section of the.ini file.

## Note

The ADV_COMPAT_FLAGS value must be an integer.

int adv_compat_flags added to hold the value of the advanced compatibility flags.

The define for rgSetupInfo[0].init_user_io has been changed to #define CPUCOMPAT_OVERRIDE B_X86_USERIO.

Code has been added so that the values of adv_compat_flags and rgSetupInfo[0].init_user_io determine the define for COMPAT_OVERRIDE in config.h.

- CF11765: The Wizard should select **Enable SoftStax** radio button by default. Furthermore, the **SoftStax (SPF) Support** and **User State Debugging** boxes should be checked by default in the **Master Builder** window.

  The fix for this problem is with the port specific.ini files, not the Wizard itself.

  The following changes were made to check the **SoftStax (SPF) Support** and **User State Debugging** boxes by default:

  1. The STARTUP key has been changed in the **SPF** section from FALSE to TRUE. This change enables the **Enable SoftStax** radio button by default.
  2. The SPF and UNDPD key in the **SUPPORT_MODULES** section have been changed from FALSE to TRUE. These changes check the **SoftStax (SPF) Support** and **User State Debugging** boxes in the **Master Builder** window by default.
  3. RAM_DISK and DD_DEVICE keys in the **RAM_DISK** section have been changed from FALSE to TRUE. This checks the **Enable RAM disk** and **Map RAM disk as /dd** check boxes.

4. The DISK_MODULES and DISK_UTILS keys in the **SUPPORT_MODULES** section have been changed from FALSE to TRUE. This checks the **Disk Support** and **Disk Utilities** boxes in the **Master Builder** window.
5. The SYS_DEVICE and DISK_IDC keys in the **INIT_MODULE** section have been changed to `/r0` and `3615`, respectively. This selects the **/dd radio** button by default.

- CF12141: The Wizard (v1.67) does not include PKMAN when it should.

  If SPF has been enabled, but the **Start inetd** option has not been selected, the Wizard does not include `pkman`, `pkdvr` and `pk` modules, required for applications using pseudo-keyboard (Hawk debugger demons, telnetd, ftpd). In this case, the Hawk demon will not be able to access **Process I/O** window.

  `build.c` has been changed to include PKMAN modules when SPF support is selected in the **Build** window.

## Known Issues

- CF7029: Configuration Wizard enhancement requests:

  1. The **Init Options** screen allows the user to set the system time zone offset from GMT. It would be much friendlier if the user could select the timezone from a list. Also, the current number is in minutes from GMT. This is confusing; what does "Use system time offset" mean?

     To obtain GMT offset for anywhere in the world, go to `http://greenwichmeantime.com/`. This task has been moved to CF11668 for future enhancements.

  2. **Beginner mode**, if the user has earlier enabled disk support, should default to enabled on the **Build** window.

     This task has been moved to CF11668 for future enhancements.

3. In the **Advanced Mode** of the Wizard only a single window may be opened at a time. Also, no Wizard menus are available when a window is opened.

   This task has been moved to CF11690 and is not presently resolved.

# Hawk Notes

The following sections represent changes and updates for Hawk 2.3.

## Enhancements

- CF9021: Add a Command I/O area.

  A Command I/O debugging window has been added that allows a user to type in commands like “step”, “next”, and “break”, as well as make a script of commands and play them back. For a full list of commands supported by the Command I/O window see Appendix B of the ***Using Hawk Manual***.

- CF9354: Add save/restore profile session.

  Two menu items have been added to the file menu. One is **Save**, which is used to save the current profiler information to a file. The other is **Restore**, which is used to open a file that has been previously saved and display the information in the profiler windows. Previously, there was a **Save** menu option; it has been renamed to **Save as Text**. Its functionality is the same.

- CF10897: **Project** -> **Load All/Unlink All**.

  **Load all** and **Unlink all** items have been added to the project pop-up menu. They will either load or unlink all of the components in a project.

- CF11634: The `.h` file is not automatically saved prior to compilation.

  Now when Hawk does a build it checks to see if all of the header files included in the project have been modified but not saved. If the **Auto-save before compilation** option is selected the appropriate header files will be saved.

- CF11753: Temporary directory.

  In the latest version of Hawk, before adding the `-td` (temporary directory) option to the command line, Hawk will check if the directory it is about to use exists. If it does not exist, the directory is created before the build command is issued.

- CF11757: Build commands.

  Previously there was no history kept in the **Build Command** window. A user could only see that last command executed. The **Build Window** will now show all of the commands executed when building a project. The first command will be at the top of the window and the subsequent commands will be added below the first in the order that they were executed.

- CF11819: Library updates to `hawkdata.xml`.

  The library section in `hawkdata.xml` that list the items appearing in the **Library Selection** dialog box has been updated as follows:

```
  <LIB name="ANSI C Library" desc="The ANSI
standard library." ocode="clib.l"
icode="clib.il"/>
```

```
    <LIB name="CPU Library" desc="CPU dependent
functions." ocode="cpu.l" icode="cpu.il"/>
```

```
    <LIB name="The non-ANSI C Library"
desc="Contains functions for compatibility with
old K&amp;R source code. It is strongly
recommended that when writing new code, a
programmer not use the sys_clib library
functions." ocode="sys_clib.l"
icode="sys_clib.il"/>
```

```
    <LIB name="Termcap Library" desc="The termcap
database screen manipulation functions."
ocode="termlib.l"/>
```

```
    <LIB name="Small ANSI C Library" desc="To
minimize code size, smaller versions of some
library functions are available in sclib.l and
sclib.il." ocode="sclib.l" icode="sclib.il"/>
```

```
    <LIB name="Unix-like Library" desc="The
UNIX-like library functions." ocode="unix.l"
icode="unix.il"/>

    <LIB name="Socket Library" desc="The BSD-like
socket library functions." ocode="socket.l"
icode="socket.il"/>

    <LIB name="NetDB Library" desc="The BSD-like
nameservice library functions." ocode="netdb.l"
icode="netdb.il"/>

    <LIB name="NDB Library" desc="The nameservice
additional library functions." ocode="ndblib.l"
icode="ndblib.il"/>

    <LIB name="ITEM Library" desc="The ITEM
SoftStax library functions." ocode="item.l"
icode="item.il"/>

    <LIB name="RPC Library" desc="Remote procedure
call library functions." ocode="rpc.l"
icode="rpc.il"/>

    <LIB name="Shared Maui Library" desc="The
standard Maui library functions." ocode="maui.l"
icode="maui.il"/>

    <LIB name="Static Maui Libraries" desc="The
static Maui library functions." ocode="mauilib.l;
mfm.l" icode="mauilib.il"/>
```

## Note

Although `mfm.l` is listed, `mfm.il` does not exist. This is why it was not included in the Static Maui Libraries listing.

- CF12300: End of line.

  This is a user preference; however, the editor should be changed to wrap the cursor at the beginning of the next line when the insert position is at the end of a line and the right arrow is pressed on the keyboard.

  A modification was made to `mwhawk.ini` that will turn off the **Allow cursor in virtual space** option. The result of this is that the cursor will now behave in the desired way.

## Resolved Problems

- CF3017: Once the **Check** or **Cross** buttons in the **Properties** window had been selected, the short-key or tab key becomes unavailable.

  This was a problem with the window not having the correct focus; it has been corrected.

- CF4192: Alternate connections to the target do not work (Profiler).

  The Profiler does not support multiple connections. Now, the second attempt to connect will fail.

- CF7673: `MWSRCDBG.dll` returns an error message with a valid breakpoint.

  If an invalid breakpoint was set, an access violation would be caused in Hawk because when it parsed the breakpoint strings returned from `mwsrcdbg.dll` it could not handle the error message. The error message is now output to Hawk's status bar and not included in the list of breakpoints which prevents the access violation from occurring.

- CF7843: If you right-click on a descriptor and select **Load**, it will not work.

  Hawk now supports the loading of file descriptors.

- CF8866: (Hawk 2.0) Building a component with sub-components fails.

  Previously, Hawk failed to build all of the components in a project when a component contained sub-components. Hawk handles the building of sub-components properly in this release.

- CF9041,10530: Hawk null modem driver `mdmnull.nt40.inf` does not work with Windows 2000. SLIP Interface setup in Windows 2000 Professional.

  The `mdmnull.nt40.inf` file has been modified to reflect the changes to the way a SLIP connection works with Windows 2000.

- CF9097: Using Windows 2000, the module properties and icons do not show up for OS-9 modules.

  This is not a Hawk problem, rather a Windows 2000-specific problem. This problem goes away when Windows 2000 have Service Pack 2 installed. There is a problem with Windows 2000 and files with no extensions (.xxx), and this problem is first fixed with SP2. For more information about this problem, go to `http://support.microsoft.com/support/kb/articles/Q265/3/26.asp`

- CF10258: Problems with Pentium4 and Win2000.

  On a very few number of Windows 2000 systems, Hawk would halt its startup process while the splash screen is displayed. The problem was due to a function call Hawk made to the system's sound device that never returns execution back to Hawk. Hawk has been changed so that this no longer happens.

- CF10355, CF10358: Hawk Debug STOP Function.

  The variable that stored whether or not a user had pressed “stop” was not being reset back to `FALSE` if the Debugger was exited after stop had been pressed. If a user reentered the Debugger, there would then be problems the next time stop was pressed. This has been fixed in this release of Hawk.

- CF10516: After exiting the Debugger, the user has to wait about 30 seconds to restart.

  In edition 68 of `spfndpdc` and `undpdc` addrmode is initialized to `1` in the `open_socket` function in `socket.c` so that the port will not have to wait for as long before it is reusable.

- CF10543: Hawk crashes when Debugger exits on error.

  The Hawk Debugger was crashing with an access violation when a program exiting with an errorcode like 216. This issue has been resolved and it will no longer cause Hawk to crash.

- CF10555: Error using the Debugger.

  There was a hard-coded size limit on the length of a function name in the Debugger code. The Debugger would crash if a function name that was read was longer than the allocated limit. The limit has been updated to be auto-resized when a larger function name is “read in”.

- CF10632: Access violation in the Debugger.

  There was a memory allocation problem with large strings in Hawk that would cause the Debugger to crash when the strings were viewed in the locals or watch window. This problem has been fixed in this release of Hawk.

- CF10646: **Set Context** in the **Symbol Browser** may cause Hawk to crash.

  Using the **Symbol Browser** to set a context that the Debugger had not yet been in was causing an access violation that would result in Hawk crashing. This has been fixed in the current release of Hawk.

- CF10647: **Symbol Browser** does not update properly after exiting Debugger.

  Many access violations would occur if the **Symbol Browser** window were left open and a user exited a debugging session. The current version of Hawk closes the **Symbol Browser** if it is open before exiting; this solves the problem.

- CF10657: Generation of I-Code libs does not work.

  When the format used to save Hawk projects changed in 2.0 the command that generated the link command for I-Code libraries was incorrectly modified. This has been fixed and the link command is now generated correctly in the current version of Hawk.

- CF11159: `mwsrcdbg.dll` access violation on board compiled for p860.

  Board ports compiled with the `-tp=p860` processor were not supported and would cause access violations when code was debugged using them as a target. Support for the 860 processor has been added.

- CF11491: Additional options do not work with old settings.

  The additional options were not being parsed correctly and if there were more than one, only the first one would work. The **Additional options** edit box can now support the specification of multiple additional options.

- CF11504: Problem switching from O-code to I-code library generation in Hawk.

  When switching the type of component, the component's properties were not being reset. In this version of Hawk, when you switch the component type the relevant properties (i.e. root psect, I-Code or O-Code link, stack checking, etc.) get set to the new component type while leaving other properties (such as include and execution) folders alone.

- CF10697,11758: Working directory. Support relative paths in Hawk.

  Hawk had been switching its working directory to the directory in which the source it was about to compile lived. This method caused problems; if the source was in different directories an intermediate directory was defined with a relative path. Hawk has been changed to keep the project directory as its working directory through all phases of the compile.

- CF11771: **Selecting Libraries** is no longer working in the Hawk2.2 update.

  The libraries selected were being added to the additional prelinker libraries instead of the O-Code libraries. This has been fixed in the current version of Hawk.

- CF11849: Using Hawk command window, cannot "di" and dissemble memory.

  Previously, the `di` command in the command I/O window would simply dump the memory, and not dissemble it. Hawk has been fixed so that this command will disassemble the specified memory.

- CF11873: Specify Process I/O font.

  The font type and font size items that were added to change the font of the source window in the stand alone Debugger will now also change the font type and size in the process I/O and command I/O windows.

- CF11893: Target environment variables.

  Before the target environment variables text box was cleared every time the connect dialog was displayed. Now, the text in the **Target Environment Variables** box has been added to the list of things that are saved and restored when displaying the connect dialog box.

- CF12137, 12243: A Critical error occurred in Hawk when using the tool tip variable evaluation. Hawk crashes on code sense feature with corrupted variables.

  This problem occurred if Hawk was trying to evaluate a variable that was NULL, using the CodeSense feature. Later on in the routine the variable that held what was to be the string representation of the variable, was deleted, and because the variable was never initialized an error was caused. A check has been added to make sure the variable being deleted is not NULL. The CodeSense evaluation now works without crashing.

- CF12165: Default name for I-Code libraries get destroyed.

  After the first build, the default name of an I-Code library would be truncated to `.il`. This was caused because components created with the I-Code library type did not have their output name stored correctly. The name was being destroyed when the `.il` extension was being added to it. This has been fixed in the current version of Hawk.

- CF12166: Changing a component from O-Code lib to I-Code library.

  When changing the component type from O-Code to I-Code, the extension of the previous type stays. When a component type was changed, Hawk was appending the extension onto the output name as if it were a new component being created, which means it was not checking to see if there was already an extension there. Hawk has been fixed so that it removes any existing extension on the output name before adding the new extension.

- CF12167: Default extensions for selection of I-Code libraries.

  Hawk was using the incorrect extension for I-Code libraries that it made when the user did not specify an extension. Hawk has been fixed so that it now uses the correct extension, `.l`.

- CF12203: Should not hard code port numbers used by `spfndpd` and `spfnppd`.

  Before `spfndpd` and `spfnppd` were hard coded to use ports 13312 and 13568. If some other application uses these ports then we can not debug or run the profiler on that target. Hawk and the Profiler have been changed on both target side and host side. Users can enter their hex or decimal port number through the new added port number dialog in Hawk or the Profiler. The port number of `spfnppd`, `spfndp` and `undpd` can be set. On the target, the port number will use the value in `inetdb` module by calling function `getservbyname` if users do not specify the port number from the command line using `-n=port number` option. (The `-n` option has been supported before.)

- CF12226: Build done modal dialog.

  After a build is complete, Hawk prompts the user with a dialog box that is always in the front that says build complete. The user then has to hit enter of click **OK** to make the box disappear. A check box has been added to the build dialog box that says "Close this dialog unless there are errors or warnings." If the check box is checked upon completion of a build, if there were no errors, the dialog box will be automatically closed.

- CF12239: Cannot set breakpoints without opening Browser window.

  When a user attaches to a module in system state, Hawk will now set the context to the module to which the user just attached. This will eliminate the need for the user to set the context to that module before they can set breakpoints in the source code.

- CF12302: Breakpoint editor problem.

  If the Breakpoint Editor is used to set a "count", the arrow keys cannot successfully be used to adjust the count by a large amount. The acceleration for the key is set too high. The arrow key described in the problem has an acceleration property that can be set from one to 10. Before it was set to five; however, that value has been changed to one (the slowest setting). If the arrow is held down, it will now increment by one slowly--about 40 times--then it will start to increment by five and eventually by 10 and 20.

- CF12314: Deleting contents of a file when building a second time.

  The code that is called when closing the Build Window declares which window is the active window to be altered. By not closing the **Build Window** and doing a rebuild when it is time to clear the errors in the **Output Window** from the previous build, the source code window was being cleared instead because it was the active window.

  Code has been added to close the **Build Window** before a new build is done. Code has also been added before the call to clear the error file that sets the active window to the **Output Window** so it is the only one that can be cleared.

- CF12317: Close the **Build Complete** window by hitting **Enter**.

  Previously, even though the **OK** button on the **Build Command** window was active, if the **Enter** key was pressed the window would not close. An event handler was added for when the **Enter** key is pressed to solve the problem.

# Hawkeye Notes

The following sections represent changes and updates for Hawkeye 2.1.6.

## Resolved Problems

- CF11452: In Hawkeye, if you selected the events data on a time line and copied it, you would be unable to paste the copied data to a new file.

  This problem has been fixed.

# TECH-CHECK Notes

The following sections represent changes and updates for TECH-CHECK version 1.1.1.

## Resolved Problems

- CF11142: Information about when and where products are installed is now stored in the registry by Microware's installers. In addition, Tech-Check has been enhanced to add the information that the incident reports, sorted by installation date.
- CF11145: Tech-Check now determines if a module set exits before trying to download it. If it does not exist, an appropriate error message is generated.

# Chapter 4: Components

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

- **OS-9 Compiler Notes**
- **PersonalJava Solution for OS-9 Notes**
- **Networking Notes**
- **MAUI Notes**
- **SNMP Notes**
- **OS-9 Utilities Notes**

# OS-9 Compiler Notes

The following sections represent changes and updates for the OS-9 Compiler version 2.4.1.

## Resolved Problems

- CF4714: The Compiler front end is aborting rather than giving a reasonable error message for the following code (which is not valid C, since the result of a cast is never an lvalue):

  ```
  inta, *b;
  b = &(int) a;
  ```

  It now runs correctly, indicating that the `unary&` operator requires an lvalue or function as operand.

- CF7624: The `-c` compiler option allows `const` qualified pointers to be kept in the code area.

  Under OS-9, this cannot be done in a standard-conforming way because the values so stored cannot be actual addresses; instead they must be offsets.

  To what the offsets are relative depends on whether or not the pointed-at object is in the code area. The MIPS compiler, when targeting a 64-bit MIPS processor, was generating code that used the wrong base address when converting one kind of offset to the other. This error has been corrected.

- CF8229: The MIPS back end contains a check for situations in which some shift operations can be avoided.

  This check was not specific enough, however, and let through some expressions in which the optimization would not work. This error has been corrected.

- CF9700: The compiler back ends attempts to minimize a weighted sum of the time and size costs of the code it generates.

  This sum was being saved in a field small enough to allow an overflow of extreme weight values in the field, which might have lead to choosing of the wrong code. The field is now larger; thus, this problem should not occur.

- CF11700: The MIPS back-end (bemips) is incorrectly sign extending unsigned values less than 32 bits when loading them from the stack. This may lead to incorrect behavior concerning the values.

  This problem has been corrected.

- CF12107: The register renaming code in optmips is reusing a register that is not free.

  This problem has been corrected; no longer is there a corruption of the semantics of original assembly code.

# PersonalJava Solution for OS-9 Notes

The following sections represent changes and updates for PersonalJava Solution for OS-9 version 3.1.

## Enhancements

- 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.
- CF9757: In addition to the MicroType fonts released in previous versions of PersonalJava Solution for OS-9, TrueType fonts are now shipped. 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`).

## Resolved Problems

- 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()`.

- 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.
- CF9128: Mouse clicks were being routed to the wrong lightweight component due to the failure to update the top component in some situations, particularly with touch screens.

  This problem has been corrected and now the event is properly routed.

- CF9560: Applets are no longer denied the use of multiple top-level Frames.
- 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.

- CF10680: High system load with PersonalJava graphic and Socket I/O.

  The Sun reference implementation of Java supports a concept called “heap reserves”. Heap reserves are intended to help the JVM respond gracefully to memory problems. These reserves can lead to puzzling behavior. For instance, if you are running a browser, you may be able to browse to a web page, view it, browse other places, then return to the page you originally viewed--only to find that some (or perhaps none) of the images display correctly. At times, there will be no warning or error messages printed explaining the condition. The AWT silently protects you from trying to display images in a low-memory state. If the JVM is later able to successfully transition back to GREEN state, the web page may once again load and display fully. Therefore, heap reserves may be useful in some circumstances, but cause trouble in others.

  The heap reserves can be controlled via two command-line options: `-my<yellow-reserve-size>` sets the size of the yellow reserve, and `mr<red-reserve-size>` sets the size of the red reserve. By default, the yellow-reserve is set to 128k and the red-reserve is set to 32k. Setting both reserves to `0` (`-mr0 my0`) effectively turns off the heap reserves mechanism.

  If you turn off the heap reserves mechanism and your application runs out of heap, Java will throw an OutOfMemory exception. With heap reserves turned off, it might still be possible to cause a browser to consume so much heap that it can no longer run, but the exception will be thrown and the application can deal it with however it sees fit.

  In this version of PersonalJava Solution for OS-9, both reserves will default to `0` bytes, and customers who wish to use the heap reserves mechanism can enable it with the command-line options.

- CF10841: In previous implementations of PersonalJava Solution for OS-9, the `rename()`function in the java_io package would return an error if the device was a PCF device and the parameters to `rename()` were not simple filenames. For instance, if `/mhc1` was a PCF device and you called `rename()` as so:

  ```
  rename("/mhc1/mydir/oldname", "/mhc1/mydir/newname");
  ```

  The above call would fail and return an error.

  The error has been fixed; the above syntax will now produce the expected results.

- CF10893: Font and height of Label.

  AWT does not show any text if the text is too small to fit in the Label. That is, if you increase the height (second parameter to setBounds) both lines will display accurately. On Windows, if you lower the size down to four, for example, it will show only the middle four pixels of each line. However, this AWT does not function like Windows'. FontMetrics is a class you can use on Windows and OS-9 to learn about fonts (like the maximum height of the characters in pixels).

## Note

It is still illegal to use the `rename()` function to move a file from one directory to another, for example:

```
rename("/mhc1/olddir/oldfile", "/mhc1/newdir/newfile");
```

The above call will fail and produce an error. On PCF devices, `rename()` can only be used to rename a file within a directory; it cannot be used to move file from one directory to another.

- CF11782: PersonalJava Solution for OS-9 version 3.0.1 crashes with an exception. PersonalJava applications that are graphics and `malloc()` intensive can crash with various exceptions.

  This was due to the lack of locking in PersonalJava Solution for OS-9 version 3.0.1 across `malloc()` calls. The delivery of a thread-swapping signal while in the malloc sub-system could cause an exception. A corrected version of PersonalJava Solution for OS-9 version 3.0.1 can be made available. PersonalJava Solution for OS-9 version 3.1 did not have problem because there was inherent locking involved with threading C libraries.

# Networking Notes

The following sections include the release notes for SoftStax/LAN Communications Pak version 3.6.

## Protocol Modules

- Thread Support

  A socket close now blocks if another thread within the same application is blocked on either a read or write.

- spip

  When booting via bootp, spip now can get the correct subnet mask from the bootp packet. Previously, it only retrieved the IP address and made an attempt to guess the subnet mask.

  In addition, spip now checks the low-level IP address if there is no present address, or if the address is `0`. If booting from bootp, low-level IP can be `0`, and high-level IP can get the IP address from the bootp server. DHCP also works. (If high-level address is `0`, it will automatically retrieve one from DHCP server.)

- spslip

  CF7474: When stopping a slip interface, it no longer hangs when attempting to terminate the slip tx and rx threads.

- spenet

  CF10443: The `SPF_GS_PROTID` getstat is now supported by spenet.

- sppscf

  CF12481: The sppscf driver now works correctly with SCF drivers that have globals.

## Utilities

- ftp

  The ftp client now ignores unknown service errors when attempting to pre-extend a file before retrieving data. This allows `/nil` to be used as a file destination without requiring the `-s` option.

- telnetd/pkman

  CF10861: NULL characters are now passed through pkman allowing telnetd to receive binary data.

- inetd

  CF7095: The `-e` and `-f` options for ftpd can now be specified in the `inetd.conf` section of an inetdb data module.

- dhcp

  CF11342: The dhcp client now works correctly with the Solaris 7 DHCP 2.0 server. Previously, requests smaller than 300 bytes were being sent and dropped by the server.

- IP Infusion

  CF11868: The "show interface" command now works correctly on interfaces other than the loopback. The "write file" command is now able to save the current configuration to a disk file. All ip infusion modules now have the option of getting their configuration from either a disk file or a data module.

## Library Functions

- The library functions `putnetent()`, `putprotoent()`, `putservent()` have been added to `netdb.h`, and documentation has been updated to reflect the changes.

## Drivers

- SPQUICC

  CF7420: The spquicc driver now correctly stops the Ethernet channel before updating the multicast hash table.

- CF10770: llbootp has a new maxbootptime parameter. By default bootp requests are initially sent 1 second apart and build to a maximum of 60 seconds. By setting this parameter, the maximum can be reduced.

- Ethernet drivers

  All cache flush/invalidate calls now correctly set the `C_ADDR` flag when flushing/invalidating a region of memory.

# MAUI Notes

The following sections represent changes and updates for MAUI 3.2.

## Enhancements

- demo_get_ucm_font() has been enhanced to read "wrong endian" fonts.

  The UCM font reader code was enhanced to detect the endianness of the data and compensate at run-time. This makes it easier to move font modules between processors.

- GFX_CM_A8_RGB888 has been added to maui_gfx.h.

  GFX_CM_A8_RGB888 is defined as 24 bit RGB with 8 bits of alpha where 0xFF is opaque and 0x00 is transparent.

- The -g[=]<name> option has been added to the demo winmgr.

  This option is used to override CDB graphic device name.

- CF2674: Create a CDB validation test utility.

  A new utility was created in $mwos/SRC/MAUI/DEMOS/CDBVAL to validate CDB modules.

- CF2898: Support needed for VT100 functions keys to mp_kybdr.

- CF11027, CF11122: msginfo -d reported error 10:45 when using not using mauidrvr_filter.

  The default MAUI configuration does not support msg_set_filter(), which is called by msginfo with the -d option. Systems may be configured with the mauidrvr_filter driver instead of mauidrvr to support msg_set_filter(). The mauidrvr_filter driver is a bit larger and slower than mauidrvr; therefore, it is not the default. To eliminate the confusion this error message can cause, msginfo was enhanced to check for this error and report to the user that they need mauidrvr_filter instead of mauidrvr.

- CF11103: Enhance jview to support reading JPG images from data modules.

  Like simg and fcopy, jview first tries to link to a data module using `<filename>` and then tries to open the file from the current working directory.

- CF12138: Add support to distinguish NUMPAD keys from other keys.

  `INP_NUMPAD` was added to `maui_inp.h` as a new `INP_KEYMOD` and `mp_xtkbd` was enhanced to set the `INP_NUMPAD` bit in the modifiers flag in key messages for NUMPAD keys on the keyboard.

## Resolved Problems

- CF7912: Several MAUI demos were bad examples for the use of `msg_dispatch()`.

  Most of the MAUI demos were not checking the return value from calls to `msg_read()` to make sure they did not return an error (specifically `EOS_SIGNAL`) before calling `msg_dispatch()`. All of the demos using `msg_read()` have been modified to make these checks.

- CF12091: `drw_polygon()` problems with filled, symmetrical, inverted (bow tie) shapes.

  While using the `DRW_FM_SOLID` drawing mode, the `drw_polygon()` did not draw identical filled polygons when two inverted shapes of the same dimensions were drawn but the defined vertexes were in different order. The polygon function was not paying attention to the fact that the line had slope when it was setting its polygon edges to fill the shape. The fill pattern is now symmetrical when the vertexes are any order for bow tie shape.

- CF12136: Problem with the threaded MAUI demo program mt_msg.

  There was a problem in the way the message read/write demo program was written. mt_msg created two threads that accessed the same system path (MAUI mailbox) on the read and the write. The write hung up in the system setstat call. The mt_msg application now uses a non-blocking read, and send signal on data ready.

# SNMP Notes

The following sections represent changes and updates for SNMP version 15.2.

## Enhancements

- CF11315: Unsupported interface statistics are now returned as zero. Additional interface types have been defined in `net/if_types.h`.

# OS-9 Utilities Notes

The following sections represent changes and updates for the OS-9 utilities.

## Enhancements

- The link utility has been changed to return an error when link is given a pathlist that does not lead to a module. Previously, if a pathlist such as `/r0/module` was used, link would complete successfully if only the r0 module existed.
- CF2466: All symbols have been forced to FLOAT before performing the main layout; all unused symbols will be thrown out and every symbols can be layout properly.

  In addition, some new PPC targets were added. This fix is reflected on Edition #65 (Rev. 1.10).
- CF2790: editmod has been modified such that if a descriptor file is not supplied on the command line (in create mode), editmod will search for `systype.des` and use it if found. If it is not found, it will search for `config.des` and use it if found. If it is not found, editmod will exit with an error.
- CF9282: mshell `print()` function sends output to `stderr` instead of `stdout`. 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 or in a script file, it will write to `stdout`. This new option is also inherited through the `_SHELLPARAMS` environment variable.
- CF9732: The following were added: `elifdef`, `elifndef`, `elifexists` and `elifexists`.

- CF11540: An enhancement was made in rename utility for OS9000. Now `filename2` can be mentioned with a full path or just with `filename`. However, the path for `filename2` should be same as the path for `filename1`.
- CF11821: The parsing routines now recognize portions of a conditional expressions that do not need to be evaluated.

# 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_________________________________

RadiSys.

MICROWARE SOFTWARE