



Microware OS-9[®] Release Notes

Version 4.4

RadiSys
THE POWER OF WE

www.radisys.com
Revision A • September 2004

Copyright and publication information

This manual reflects version 4.4 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 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.

August 2004

Copyright ©2004 by RadiSys Microware Communications Software Division, Inc.
All rights reserved.

EPC and RadiSys are registered trademarks of RadiSys Corporation. ASM, Brahma, DAI, DAQ, MultiPro, SAIB, Spirit, and ValuePro are trademarks of RadiSys Corporation.

DAVID, MAUI, OS-9, OS-9000, and SoftStax are registered trademarks of RadiSys Microware Communications Software Division, Inc. FasTrak, Hawk, and UpLink are trademarks of RadiSys Microware Communications Software Division, Inc.

† All other trademarks, registered trademarks, service marks, and trade names are the property of their respective owners.

Contents

Chapter 1: Introduction

Chapter 2: Operating System

Resolved Problems.....	8
------------------------	---

Chapter 3: Host Applications

Configuration Wizard Notes.....	12
Resolved Problems	12
Hawk Notes	12
Resolved Problems	12

Chapter 4: Components

OS-9 Compiler Notes	16
Enhancements	16
Resolved Problems	16
Networking Notes	18
Enhancements	18
Resolved Problems	19
OS-9 Utilities Notes.....	19
Resolved Problems	19
MAUI Notes.....	20
Resolved Problems	20
Add-Ons.....	20
Resolved Problems	20

1

Introduction

Microware OS-9 version 4.4 represents a maintenance and update release to incorporate all of the improvements that have been introduced into the component parts. In addition, this the first release to officially include support for MIPS64 processors.



MIPS64 Users:

This is the first official release of OS-9 for MIPS64.



It is recommended that you read all the release notes, starting from any previously used version of OS-9 to these notes for version 4.4.

2

Operating System

This chapter provides an overview of the changes and improvements made to OS-9 since the last release.

For information about...	Go to this page...
Resolved Problems	8

Resolved Problems

The following section describes CustomerFirst incidents related to the OS-9 operating system and how they were resolved for the current release.

- CF16192: `_os_ev_wait()` does not return the signal code if a signal is already queued.

When doing an `_os_ev_wait()` with a signal pending, `_os_ev_wait()` would return with `EOS_SIGNAL`, but the signal value would return an undetermined value. `_os_ev_wait()` now returns the value of the last signal in the user signal queue when returning with `EOS_SIGNAL`.

- CF16259: The SH7750R port should not contain its own `flshcach.c` source file

The SH7750RSE port directory was modified to no longer have a private version of the normally shared source file. The global `flshcach` module now identifies the specific SH-4 processor at run-time and flushes caches properly.

- CF16347: PCF problem deleting files with long file names if full path was specified.

PCF had a problem deleting files in a directory when multiple files with long file names were in a directory and a full path name was specified. The problem only showed when the directory slots consumed over two clusters. This problem has been resolved.

- CF16527: PCF for OS-9/68K returns #000:214 (File Not Accessible) error when trying to open file with the `FAM_APPEND` mode bit set.

The PCF file manager was improperly detecting file access when append mode was specified. This has been resolved with edition 84 of PCF for 68K

- CF16531: FPU-using threads cause a memory leak when they terminate

Due to a problem in some processors' FPU modules (MIPS, PowerPC, x86, and ARM) user-state memory would be lost when threads that had used the FPU terminated. This problem has been corrected.

- CF16542: Enhance `pcilib.1` and `cpicirq` to work better on PowerPC 8xx systems

`pcilib.1` and `cpicirq` have been modified based on suggested information in the Users Manual. `pcilib.1` now masks interrupts while modifying the `CIMR` register. `cpicirq` interrupt masking was enhanced.

- CF16556: `cache403` and `flshcach` for the PowerPC 403 only flush 1/4 of the cache.

It was discovered that the `cache403` and `flshcach` modules for the PowerPC 403 only flushed one quarter of the cache. These modules have been updated to flush all of the available cache.

- CF16559: Possible CPU exception in `kernel` on SuperH

An exception could happen on SuperH when returning from an illegal service routine request. This problem has been resolved with `kernel` edition 177 for both SH-3 and SH-4.

- **CF16597: Multiple threads doing `_os_chdir()` can cause system corruption**
`ioman` was modified to ensure that only one thread within a process is allowed to perform a directory change for the process at any given time. In general, the use of `_os_chdir()` and relative paths must be carefully architected for threaded processes.
- **CF16597: Threaded processes can possibly close system path number 1**
`ioman` was modified to ensure that the translation from user path number to system path number doesn't incorrectly allow access to system path 1. In general, the use of user path numbers must be carefully architected for threaded processes.

3

Host Applications

This chapter contains release notes for host applications used with OS-9 v4.4.

For information about...	Go to this page...
Configuration Wizard Notes	12
Hawk Notes	12

Configuration Wizard Notes

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

Resolved Problems

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

- CF16554: Certain port directories cannot be built with the Wizard using the "Build Port" button

The effected ports have been corrected by removing the incorrect `[BUILD_PORT]` group in their `.ini` files. The default values build into the Wizard will work correctly.

Hawk Notes

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

Resolved Problems

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

- CF13763: Add missing PowerPC boards to chip selection in Hawk
The MPC 555, MPC 750, MPC 8240, MPC 8260, and PPC 405 processors were not available when creating a component. These processors have been added to the list of available processors for PowerPC.
- CF15675, CF16120, CF16231: Hawk help no longer comes up when F1 or Alt-F1 are pressed.

CodeWright has removed support for the API Assistance that came up when Alt-F1 was pressed. As a result, Hawk no longer has this functionality.

Hawk now ships with the following help files found in the `MWOS\DOS\BIN` directory:

- `compiler.chm`
- `os.chm`
- `gfx_audio.chm`
- `network.chm`

When a function is highlighted with the mouse and F1 is pressed, the associated help file will be opened with help describing the function.

- CF16501: Hawk does not properly open source files to lines with errors

If errors were discovered when compiling in Hawk, double clicking on an error line or using the "Goto next message line" button in Hawk would not always

go to the line containing the error and would report "Unable to locate file" on the status bar. This was caused by the use of two error parsers in Hawk. `_MicrowareErrorInfo` is the only error parser used in Hawk now, and it will properly open the source file to the line containing the error.

- CF16502: The Hawk debuggers print that signal zero was received when it was really non-zero.

The Hawk debuggers now correctly print the signal value that the remote process received. This was only a problem when the remote process did not have a signal handler.

- CF16679: Hawk does not regenerate dependencies when relative paths are used

Hawk did not regenerate dependencies when relative paths were used, and the current working directory (cwd) was not set to the project directory. Hawk now searches for files relative to the cwd and the project directory.

- CF16701: Hawk changes the current working directory (CWD) when adding a library.

Hawk has been modified to not change the CWD when adding a library to a component. Requires HawkPrj.dll v2.3.4.11.

4

Components

This chapter contains processor-independent release notes for OS-9 components.

For information about...	Go to this page...
OS-9 Compiler Notes	16
Networking Notes	18
OS-9 Utilities Notes	19
MAUI Notes	20
Add-Ons	20

OS-9 Compiler Notes

The following sections contain release notes for the OS-9 Ultra C/C++ compiler.

Enhancements

The following list describes general enhancements made to the Ultra C/C++ compiler for this release. Where applicable, CustomerFirst incidents are included.

- The message queue facility was expanded to include inter-process communication.

The `mq.1` library was enhanced to include support for message queues that allow one process to communicate with another. Previously, message queues were limited to being within a single, threaded application. Now, a message queue whose name begins with slash (/) will be visible to all processes and threads on the system. Inter-process message queues support virtually all the same features as inter-thread message queues: POSIX compliance, `_MQ_O_IMMEDIATE` and `_mq_notify_write()`. Inter-process message queues do not support `SIGEV_THREAD` notifications.

The inter-process communication is facilitated by a new file manager (`msgman`) and device descriptor (`mq`). This is a general-purpose message passing file manager; similar in functionality to `pipeman`. It supports both named and unnamed message queues.

- CF16377: Ultra C/C++ does not accept `switch` statements with 64-bit expressions

Ultra C/C++'s 64-bit integer support has been enhanced to include support for 64-bit `switch` statements. They are implemented internally using 32-bit constructs.

- CF16586: The Ultra C/C++ front-end is adhering too closely to the ANSI standard.

The front-end has been relaxed in these ways:

- The `long long` data type is now allowed in all modes (K & R, ANSI extended, and ANSI) without any diagnostic messages. The diagnostic messages can be restored by passing an option to the front-end via the executive: `-fe-diag_[warning|error]=450`
- C++ style comments (`//`) are now allowed in C code unless `--cpp_comments_notallowed` is passed to the front-end.
- Empty translation units (source files) are now allowed in all three modes. The diagnostic messages can be restored by passing an option to the front-end via the executive: `-fe-diag_[warning|error]=96`

Resolved Problems

This section describes CustomerFirst incidents related to the Ultra C/C++ compiler and how they were resolved for the current release.

- CF16205: Constant pointer (CP) points to wrong location

When a child thread was linked with CSL and forked on processors with a CP, the CP would incorrectly point to the CSL code rather than the user module. `mt_csl` has been modified to properly calculate the CP when a child thread is forked.

- CF16260: I-code and CSL linked, threaded applications that use `intercept()` can not get signals

If an application is I-code linked with the threading CSL I-code libraries and does not call the `signal()` function explicitly and contains a reference to a function that has a reference to `_glob_sem` (e.g. `strncpy()`) then arriving signals would crash the application. This problem in `mt_csl.il` has been corrected.

- CF16368: `cpfe.exe` gets an application error when compiling extremely large files.

When compiling very large files, `cpfe.exe` could get an application error. This was caused by an overflowed number (negative) being used as an index number of an array). The number is now kept within bounds. `cpfe.exe` edition 51 resolves this problem.

- CF16427: SH-3/4 `switch` statements don't work when they contain large negative cases

The SH-3 and SH-4 back-ends were generating incorrect code for `switch` statements containing large negative numbers. This has been corrected.

- CF16522: Ultra C/C++ can use unpredictable instructions on ARM processors

The ARM back-ends (`bearm.exe`, `bearmbe.exe`, and `bearmv5.exe`) were fixed to no longer use a register as both a writeback base register and a listed register in `ldm` or `stm` instructions (e.g. `ldmia r7!, {r7-r8}`).

- CF16530: `pthread_kill()` cannot be used again after `longjmp()` is used to exit signal handler

The `pthread_kill()` function was fixed so that if a thread sends a signal to itself and uses `longjmp()` to exit the signal handler then `pthread_kill()` can be used again. Previously a lock was held such that no thread could return from `pthread_kill()`, among other things like `pthread_exit()`, `pthread_join()`, etc.

- CF16581: Exception related signal reception can cause application to crash

The C libraries were fixed to ensure that the mechanism used to generate exception related signals for a process does not crash. Previously, it did not work correctly if the exception happened with a different global data pointer that the main application (e.g. in the `csl` module).

- CF16583: `dup2()` causes the calling program to crash if it has more than 32 paths

The `dup2()` function was fixed to correctly reallocate an internal buffer when more than 32 saved, open paths are required to perform the `dup2()`.

- CF16617: `stwcx` instruction errata for PowerPC 405

As per the errata, the `_os_sema_p()` and `_os_sema_v()` code was modified. Before the `stwcx` instruction, the `dbct` instruction was used. This should ensure the cache's store access will work correctly if an interrupt were to occur during the `stwcx` instruction.

- CF16634: Mutex problem with C++ libraries in threaded applications

It was discovered that the standard C++ libraries had a mutex problem with threaded applications. If an application using mutexes created a new mutex at the same location of a previously used mutex that was not destroyed and still contained a valid sync code, the new application would not initialize the mutex, and could try to use pointers in the mutex that were invalid. This resulted in random corruption and invalid data references.

A semaphore has now been placed around the mutex initialization, and threads competing to initialize the mutex will clear the structure before initializing the structure.

- CF16678: The Ultra C/C++ front-end crashes if `_asm()` statements appear in a namespace block

The front-end was fixed to not generate a protection fault in this situation. Further, it was fixed to not generate repeated duplicate `_asm` symbol errors in this same situation.

- CF16693: The `offsetof()` macro cannot be used in a constant integral expression in strict ANSI mode.

The `stddef.h` header file was modified to use a built-in compiler operator `__INTADDR__(x)` that takes an expression (possibly containing address-of or pointer casts) and returns a `size_t` value for the resulting constant. Now, `offsetof()` may be used as a bitfield size, enum initializer, array dimension, or case label in strict ANSI mode.

Networking Notes

The following sections include the release notes for the current versions of SoftStax and LAN Communications.

Enhancements

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

- CF16699: Add gratuitous ARP support

Edition 63 of `spenet` has been enhanced to support gratuitous ARP. A gratuitous ARP is sent out when a new IP address is assigned if the `enet` descriptor has defined `GRATUITOUS_ARP` to be 1.

Resolved Problems

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

- CF16532: Taking the SLIP interface down does not free all the memory for 68K
The `spslip` driver was fixed so that the kernel could correctly reclaim all the memory allocated when the `spslip` system-state threads were terminated.
- CF16616: System reboots when taking down PPP stack for 68K
When deinitializing the 68K PPP stack, a privilege violation exception (8) would occur, causing the system to reboot. The problem was found and resolved in the `sphdlc` driver.
- CF16856: UDP errors can go unrecognized by the IPv4/IPv6 networking stack, causing unnecessary DNS time-outs
If, for example, one attempts to use an existing machine as a DNS server, but the machine does not accept packets to the DNS port the `recvfrom()` that the resolution library does is forced to time-out rather than returning an error right away. `spip` was modified to properly indicate the error in a timely fashion. This allows the resolution library to skip to the next DNS server quickly.

OS-9 Utilities Notes

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

Resolved Problems

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

- CF16332: `mshell` redirection with `-x` problems
If the `-x` option was specified for `mshell` to exit on error, and there was a redirection error, `mshell` would print out the error, but not exit. `mshell` now properly prints the errors and exits. `mshell` was also fixed to properly report when it could not redirect input to a procedure file.
- CF16337: `mshell` can't handle an `assign` with a `-?` in it
When running a script or launching a new `mshell` with an alias containing "?", `mshell` would print the `mshell` options. `mshell` has been fixed to properly parse options and environment variables when starting a new `mshell`.
- CF16612: `mshell` does not exit if telnet connection is killed.
If a telnet connection was not closed cleanly, `mshell` would remain on the system. `mshell` now verifies the input path is still active when waiting on input every 30 seconds. Power management users can use the `-ny` option to disable this functionality.
- CF16689: `editmod` can give errors if `#undef` is used

When `#undef` was used to remove known symbols, `editmod` would remove additional identifiers. This problem has been resolved.

MAUI Notes

The following sections contain release notes for MAUI.

Resolved Problems

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

- CF16770: MAUI message API usage is blocked when a thread blocks in `msg_read()`

The MAUI mailbox drivers (`mauidrvr`, `mauidrvr_filter`, and `mauidrvr_lock`) were fixed to no longer prohibit other threads from entering the MAUI message API when a thread was blocked in `msg_read()`.

Add-Ons

This section describes the enhancements, resolved problems, and known issues for the add-on products in this release. CustomerFirst incidents and workarounds are provided, where applicable.

Resolved Problems

- CF16243: HawkEye crashes PPC403/405 system. Virtual interrupts need to be moved for PPC.

`s1m` on PowerPC used the `mftb` (move from time base register) instruction, which is not supported on the 403 or 405 systems. A new version of `s1m` module, `s1m403`, has been created for these boards. Virtual interrupts were moved from 0x40 to 0x20 for `s1m` on PowerPC.

- CF16571: PersonalJava incorrectly computes the width of characters when `charsWidth()` is used on a string containing non-ASCII character values.

PersonalJava has been updated to correctly compute the same width if either `charsWidth()` or `stringWidth()` is used. Previously, `charsWidth()` would count non-ASCII characters as width zero.