OS-9 USERS GROUP
"DEDICATED TO EXCELLENCE IN OS-9 COMPUTING"

MOTD

Officers

President:	Boisy G. Pitre
Vice President:	Carl Kreider
Secretary/Treas:	Scott McGee
Librarian:	Alan Sheltra
MOTD Editor:	Scott McGee

Summer 1992 The Official Voice of the OS-9 Users Group Volume 1 No. 1

In With the New!
(But, NOT Out With the Old!)

An Editorial by Scott McGee

This is the premier issue of the OS-9 Users Group's MOTD, and, before someone says, "but the OS-9 Users Group has published MOTD before," let me set the record clear. The OS-9 Users Group that is now in existence is not the same OS-9 Users Group. The old Users Group died out. However, with new machines and new people, the time was ripe for such a group. In response a NEW Users Group was born. It is a new group with the same old name. It is made up of new people, but with many of the same old ideals. It is a new organization but dedicated to some of the same old goals. It has a new publication that uses the same old name with a new dedication to spreading information.

No, we didn't take the old organization and try to rebuild it. We formed a new organization with all that we found good from the old, and reused the old name that meant so much to so many. We hope to live up to, and exceed, the expectations that people had of the old group. We hope to provide many of the same services, but with an eye to today's world of OS-9. We plan to support OS-9 in all its forms, 6809, 680x0, and OS-9000. Most of all, we plan to make the phrase **"Dedicated to Excellence in OS-9 Computing!"** not just a catchy motto, but a philosophy to live by.

In MOTD you can expect to find such things as:

Programs -- all languages, all versions of OS-9, all hardware.

Information -- useful information regarding OS-9, the machines that support it, and the people who use it.

Articles -- articles on interesting topics related to OS-9

Help -- Question and Answer columns by OS-9 experts.

Vendors -- the latest information on software, hardware, and where to find it.

And much more.

I hope you enjoy it!

Scott -- MOTD Editor

How To Reach us:

Mail:	Phone:
OS-9 Users Group	Scott McGee
PO Box 434	MOTD Editor
Farmington, UT 84025	(801) 583-2644

The Move to OSK

By Mark Griffith

I was slightly apprehensive when I received my MM/1 and fired it up for the first time. I had been using OS-9 for years, but wasn't sure what this new "industrial strength" version of my favorite OS would be like. I don't know what I expected, but my fears were soon gone when I slowly realized that this was still my old friend OS-9, just in a faster, more capable package. First, I tested the portability of some source code written on the CoCo. Bringing a simple BASIC09 program into the Microware Basic (the new name) editor worked and it compiled and ran without errors. I then tried a more complex program and it also worked. Pretty good so far.

Next, I tried some simple "C" program code. The compiler started spitting out errors I had never seen before. Well, I thought, that takes care of the portability theory. But, after looking harder at the errors, I realized that the compiler or the OSK operating system was not at fault. It was I, the "C" programmer, that was in error. I had become so used to the way things were done under OS-9/6809 and the "features" of that "C" compiler, that I had started to write sloppy, non-portable, code. A little editing on the source code and it now compiled and ran fine. I was a wiser person, thanks to the Microware compiler and its ability to catch errors the old 6809 compiler would let go. Not only was this new OS-9 faster, but it helped me become a better "C" programmer.

There are some other differences between the two C compilers that might cause some head-scratching to programmers moving from OS-9/6809 to OSK. With the 6809, the size of an 'int' is 16 bits. However, an OSK 'int' is 32 bits long. This is significant because many 6809 C programmers depend upon the 16 bit size for many things that would cause any code ported to OSK to not work. Another problem is with null pointers. These usually occur as a result of a programmer not initializing a pointer before its first use. With OS-9/6809, the program would normally run with no problems. That same program ported to OSK will result as a bus trap error and the program terminating as soon as the null pointer is reached. Not only is this problem easy to cause, but it can be difficult to debug and repair.

OSK provides the programmer with a wealth of functions and system services in excess of what OS-9/6809 could do. Many times, a 6809 programmer would need to keep accurate time of events occurring during a program run. Under the 6809, the smallest timing one could easily obtain was in one second intervals. Even then, the intervals were not guarantied to be EXACTLY one second apart. OSK, on the other hand, provides system calls that allow the programmer to reach a granularity in timing down to 1/200ths of a second (or 5 milliseconds) with complete confidence that it is accurate (meaning system processing will not effect the timing).

System calls for setting alarms, sending signals by date, time, or at predetermined intervals, creating events, the ability to mask signals and have them automatically queued, and setting up user accounting are just some of the many extras OSK provides. In addition, SCF devices support more features like the ability to send signals upon DCD loss, or enabling and disabling hardware flow control. SCF devices also support true non-sharable mode so only a single process can access a serial port at one time.

A complete environment is now available under OSK. Environment variables can be set at startup or within shell scripts and then accessed from within your program. Things like the default printer port, the port the modem is on, and so on can be set and accessed easily. Of course, no real standard exists for what variables should be used all the time and what their names should be so there is still some uncertainty there. I believe that as more and more applications start coming out for OSK, these environment variables will be used more often. A final note on this: environment variables can be passed to forked applications by using the os9exec() function. While this is a little tricky to setup and use properly, it makes forking of programs from within another much easier when variables need to be passed.

OSK Version 2.4 now supports variable sector size I/O on RBF devices. Previous versions of OSK and all versions of OS-9/6809 were hard coded to read and write 256 byte sectors to floppy disks (and hard disks in the case of OS-9/6809). Allowing the sector sizes to be settable gives OSK the ability to easily read and write other OS disk formats, like MS-DOS and early CP/M disks, without going through all the trouble that was needed under OS-9/6809. With the PCF device manager supplied by Microware, reading and writing MS-DOS disks is as simple as OSK disks. Many of the standard OSK utilities will also work on these disks.

On the other hand, there are some differences between OS9/6809 and OSK directory structures that can cause some problems for those making the move up. For example, the way the RBF file system works is slightly different....different enough to where utilities written for OS-9/6809 will not work under OSK if those utilities are either hard coded for 256 byte sectors, or directly read the raw RBF device and seek to specific logical sectors. With OS-9/6809, a directory entry consisted of the filename and a pointer to the LSN of that file on the disk. Seeking to that LSN would place your file pointer over the file descriptor sector of that file. Not so under OSK. Because of the difference in the sector size used in OSK (normally 512 bytes instead of 256), the programmer must read the PD_SSize byte in LSN0 to get the number of 256 byte groups in a sector and then multiply the directory entry pointer by the size byte to get the absolute physical location of the file on the disk. OSK filename entries are also 28 characters in length compared to 29 for OS-9/6809, and the OSK filenames in the directory entry need not have the high bit set on the last character in the name.

OSK SCF device descriptors also include more information than their OS-9/6809 equivalents that programmers need to know to port code from the 6809. Additional fields include the tab character, tab stops, and different codes for the baud rate byte. If OS-9/6809 code was written properly and used the scfbuf->field convention for accessing descriptor values, you should have no problems porting.

These are just some of the differences between OS-9/6809 and OSK that I have come across in the past couple years. None of them make OSK any harder to learn and use. Application writers will love the many additions that will make program development much easier, not to mention a much faster machine to work with. All-in-all, the move to OSK is not nearly as frightening as one might expect it to be.

CRLF.c

A conversion program by James Jones

The following is a highly special purpose program that I use to switch EOL from LF to CR when I've snarfed Nuxi [Unix] text files over to OS-9. NOTE: we're talking updating the files *in place*. This is a Good Thing if that's what you want, because it minimizes flailing about with deleting and renaming and avoids a (perhaps trivial) amount of fragmentation of the disk, but it's a Bad Thing if that's *not* what you want, in which case copy first, then crlf.

James Jones

Editors Note: In the following code, lines that are too long to fit within one of our columns are "wrapped", and have a backslash ("\") at the end of the first line to indicate that the program line continues on the next physical line. Most C compilers should accept the program exactly as shown here. If yours does not, remove the backslash and append the following line to the end.
Happy Programming!

```
/*
 * crlf -- a bulk-mode LF->CR translator for RBF
 * files
 *
 * Hack to use two paths courtesy Peter Dibble;
 * it is best in any case to read/write multiples
 * of a sector, but given\ the use of the two
 * paths, it is probably even more important
 * because of record locking.
 *
 * -? option added by Scott McGee.
*/
#include <stdio.h>
#include <modes.h>

main(argc, argv)
int   argc;
char  *argv[];
{
int  i;
int  inpath, outpath;

  /* if no filename is given show help */
  if (argc == 1) ShowHelp();

  /* if arg starts with '-' show help */
  for (i = 1; i < argc; i++)
    if (argv[i][0] == '-') ShowHelp();

  for (i = 1; i < argc; i++) {
    if ((inpath = open(argv[i], S_IREAD)) == -1\
||
      (outpath = open(argv[i], S_IWRITE)) == -1)
      fprintf(stderr, "crlf: can't open %s\n", \
argv[i]);
    else
      DoCRLF(inpath, outpath);
    if (inpath != -1)
      close(inpath);
    if (outpath != -1)
      close(outpath);
  }
}

#define HUNKSIZE  (10 * 1024)

char  Hunk[HUNKSIZE];

DoCRLF(inpath, outpath)
register int  inpath;
register int  outpath;
{
  register char  *HScan;
  register int   Qty;

  while ((Qty = read(inpath, Hunk, HUNKSIZE)) >\
 0) {
    for (HScan = &Hunk[Qty]; --HScan >= Hunk; ) {
      if (*HScan == '\l')
        *HScan = '\n';
    }
    write(outpath, Hunk, Qty);
  }
}


ShowHelp()
{
    fprintf(stderr, "crlf - a bulk-mode LF->CR \
translator for RBF files\n");
    fprintf(stderr, "  Usage:  crlf <path> \
[<path>]\n");
    fprintf(stderr, "          crlf -? = Show \
this help message\n");
    exit(0);
  }
```

Talking with the President

An Interview with Boisy G. Pitre by Scott McGee

I had a chance to talk with Boisy and ask him a few questions about his feelings concerning the OS-9 Users Group. I have included below the responses he gave me to questions I thought readers might like to hear.

Tell me about the OS-9 Users Group

The OS-9 Users Group is a self-sufficient organization dedicated to the growth and expansion of the OS-9 operating system. We BELIEVE in OS-9. Our officers are all experienced in the operating system and have equitable ability to provide the professionalism that OS-9 deserves.

Why start another (new) OS-9 Users Group?

The OS-9 Users Group is a revival of the *spirit* of the old Users Group. In its time, the old group served well. Now, there are new systems that promise to be excellent choices for OS-9 users, and this growing base will need a professional organization to represent it.

What purpose will the Users Group serve?

Our primary goal is the dissemination of information concerning the OS-9 operating system. As our theme says, we are "dedicated to excellence in OS-9 computing." The OS-9 Users Group also gives OS-9 users a 'home base' to meet other OS-9 users and share information with them.

What is the likely audience for the Users Group?

Just about every professional OS-9 user out there! Engineers, students, programmers, casual users... because we have something to offer for everyone, from OS-9/6809 to OS-9000, everyone benefits.

What systems do you support? (hardware, software, revisions)

The Users Group has made it a point to support all versions of OS-9, from OS-9/6809 to OSK to OS-9000. Our goal has been to encompass every OS-9 user, and offer them practical information on their particular version of OS-9. As for hardware, we will only support systems as they relate to OS-9.

How is this Users Group different from other users groups?

The OS-9 Users Group is a professional organization. I don't think that we are different from the OS-9 Community Network or foreign user groups as far as the mission goes, but we do have different means to our goals.

Will you have software?

The OS-9 Users Group Librarian, Alan Sheltra, is putting together a library of software disks which will be available to our members. And with new software being added regularly, the library is expanding.

Is the newsletter available to non-members?

The MOTD is available to paid members of the OS-9 Users Group. However, copies of our premiere issue may be shipped with computer systems from Frank Hogg Labs and possibly with other machines.

How do I join?

Becoming a member is easy! Just pick up an application from any one of the many available information services, or write to:

The OS-9 Users Group
P.O. Box 434
Farmington, Utah 84025

with your name, address, system type, and diskette size/format. Along with this information, all that is needed is a check or money order for $25.00 to secure your membership for one year. Upon receipt, your membership will go into effect immediately.

What do I get when I join?

Along with your one-year subscription to the MOTD newsletter, you will get an introdisk for your system from the Users Group Library.

How do I contribute programs?

You may contribute programs to the OS-9 Users Group Library by sending your program and documentation to the above address in care of Alan Sheltra, Users Group Librarian. It will be evaluated and, if it merits, be included in the OS-9 Users Group Library. Articles for the MOTD are also welcomed.

OS-9 Tech

Our regular question and answer column

This column is for technical questions. While most of our questions in this issue have been taken from various electronic forums, you are encouraged to submit any questions you have. We'll find an answer for you, and print those of general intrest. This issue's answers are provided by Kevin Darling, one of the Users Group's Technical Advisors.

Q I am interested in purchasing a CD drive for my NeXT computer. Since photography is my avocation, and since the Kodak Photo CD's are only fully functional on XA drives (like those used on CD-I machines), I would like to get info about these machines. Specifically, I would like to know if I can use any of these machines for a general purpose CD to also read the standard formats that exist today (9660, High Sierra, Rock Ridge) and access files on all of these CD's from my NeXT, either via ethernet or some other connectivity scheme.

A The current answer is "no". They're meant mostly as standalone players, much like a personal or living-room CD-DA player. After all, they have a computer already built into them. Yet there's so much interest in what you asked, that I do think some kind of combo model with a slave SCSI port would be popular.

A cautious answer is "maybe". There _are_ industrial CD-I models which can be expanded with Ethernet ports, but I'm not sure that there's any software available to allow using one as a CD-ROM server. You might try calling 1-800-CDI-5484 and inquiring if this is even possible.

But basically, it'd make more sense right now to get a CD-I unit for your living room, and get an XA and multi-session capable drive for your computer. Their design purposes are just too different from each other right now.

PS: ISO9660/etc. are directory layouts. I.e.: not hardware dependent.

Q I downloaded one of the fli files from the ftp site, hermit.cs.wisc.edu. It is called hp-kill.fli.lzh. It is large enough that it took quite a while to download at 1200 baud. Now I find that lharc will not extract it! lharc will successfully report that the file is in there but nothing I do will get it out!

A It sounds like it's because of the archived filename "hp-kill" with the "-" in it. A later version of lharc converts those characters.

OS-9 unzip/etc. utils sometimes have this problem as well. The easiest way around it temporarily is to use dEd or write a program that changes the "-" in the filename (it's right at the front... dump the file and see) to a usable OS-9 filename character.

What I usually do is change the "-" ($2D) to a "." ($2E) and then subtract $01 from another title character (example: "h" ($47) to "g" ($46). That keeps the dearchiver from reporting CRC errors.

So change the "hp-kill" to "gp.kill" and you should be able to break it out.

Q I have an Atari ST with a Supra hard disk drive. I have OS-9 for the machine, but it only works on floppy disks. I can setup the hard disk partition table using the OS-9 FDISK program (which runs under GEM), but when I try to format the hard disk under OS-9 (format /h0fmt), the system hangs. It appears to be waiting for an interrupt or something.

Does anybody have a modified driver that will work with the Supra drive? (well, actually it's the Supra controller that appears to be the problem; I have tried both the original Miniscribe drive and a newer Quantum drive, both to no avail)

A I think there was a Pipelines article on this, but I can't seem to find it. Anyway, this is the Supra patch for sthd ed 13 (by doing a comparison)...

```
— original
< 0x000005FA+r7>082E000207C4     btst.b #2,1988(a6)
< 0x00000600+r7>6606             bne.b 0x608+r7->
< 0x00000602+r7>003C0001         ori.b #1,ccr
< 0x00000606+r7>4E75             rts
< 0x00000608+r7>08930007         bclr.b #7,(a3)
< 0x0000060C+r7>08AE000207C4     bclr.b #2,1988(a6)
— supra
> 0x000005FA+r7>08B90007FFFFFA11 bclr.b #7,(0xFFE525B1+r7).l
> 0x00000602+r7>08AE000207C4     bclr.b #2,1988(a6)
> 0x00000608+r7>6608             bne.b 0x612+r7->
> 0x0000060A+r7>4A2E07C4         tst.b 1988(a6)
> 0x0000060E+r7>671A             beq.b 0x62A+r7
> 0x00000610+r7>60E2             bra.b 0x5F4+r7
```

Q I've noticed some "weird" behavior using OS-9 pipes. I used a command like

```
$ myprog ! grep pattern ! less
```

where I used 'less' to look at some of the output and then I quit less. After this, I found that 'myprog' and 'grep' were still running. Unlike Unix, OS-9 simply blocked the grep process when the output pipe filled up, which in turn let the output of myprog fill up that pipe.

Apparently this is because OS-9 pipes are single-ended. i.e. there is only 1 file descriptor when you make a pipe and you can use that for either read or write, therefore pipeman can't tell when there are no readers for the pipe.

A Part of the weirdness may be due to grep itself :-) If the pipe simply filled and blocked, then "myprog" and "grep" _should_ fall asleep, yah?

However, check your procs output. On my machine both programs continue to rack up cpu time. This almost certainly means that the grep I use, at least, is _ignoring_ the output write errors!! (poor C code?)

Replace grep with a different program, and I'll bet things operate more as expected. You still have to do a couple of "w" (shell F$Waits) commands by hand, though... whereas a smarter shell might keep better track of all its kids and do that for us automagically.

I think pipeman keeps track of reader/writer counts.

There is still an anomaly of some sort, however. I wrote a program which generated random chars, and another which simply passed stdin to stdout. I then did a "RANDOM ! PASSTHRU ! MORE" and killed off "more". PROCS showed MORE was gone, and RANDOM and PASSTHRU as asleep... but I would've expected the closing of MORE's stdin path to have caused Pipeman to wake up and abort PASSTHRU (and later, RANDOM). I had to do a "w" command before this chain effect actually happened. Strange. though it might have something to do with my commands being in Basic (another C program :^)

Q How do OS-9 windows _work_? I already have the screen codes and stuff from the back of the manual. But how does the system handle the windows. I know how to use the codes, but how does the system know how to use the codes? What modules do what? If I wanted to rewrite the device drivers for some modules, such as the ramdisk, or the printer, or what have you, I know what to do. But there is no information on how the windows work.

A It's actually just a very complicated SCF device, broken into pieces. CC3IO has the normal SCF driver entry points. It may call WindInt (or VDGInt) to process (INTerface) graphics escape/stat codes. WindInt may call GrfDrv to actually put text/graphics on a window-type screen.

The trouble is, it would take a mountain of docs to explain it in detail. There are around 44 entry points to the grfdrv co-driver!

The windowing isn't in the same simple class as a disk or serial port driver. Its size alone (much larger than the kernel!) points this out. And unlike other OS-9 drivers, it wasn't expected that users would need to write a different version for non-Tandy hardware.

If you really, seriously want to change the windowing (and/or learn how it works in intimate detail), your best bet is to disassemble and comment it. (Several people did this; my own disasm/comments comes to over 500K of text. Contrast that to 10K for a hard disk driver, or even 45K for os9p2. The windowing code is _very big_ :-)

MOTD
OS-9 Users Group
PO Box 434
Farmington, UT 84025

OS-9 USERS GROUP

"DEDICATED TO EXCELLENCE IN OS-9 COMPUTING"

Late Breaking Information!

To find out where, in your area, you can see a CDI demonstartion, call 1-800-223-7772

We are pleased to anounce that members of the OS-9 Users Group will receive a 15% discount on purchases made from ColorSystems. We will continue to add to the list of companies offering such discounts.

For more information on ColorSystems' products write to:
ColorSystems
PO Box 540
Castle Hayne, NC 28429