SCIENCE AND ENGINEERING RESEARCH COUNCIL
RUTHERFORD APPLETON LABORATORY

COMPUTING DIVISION

DISTRIBUTED INTERACTIVE COMPUTING NOTE 750

PERQ UNIX IMPLEMENTATION NOTE # 47   Issued by
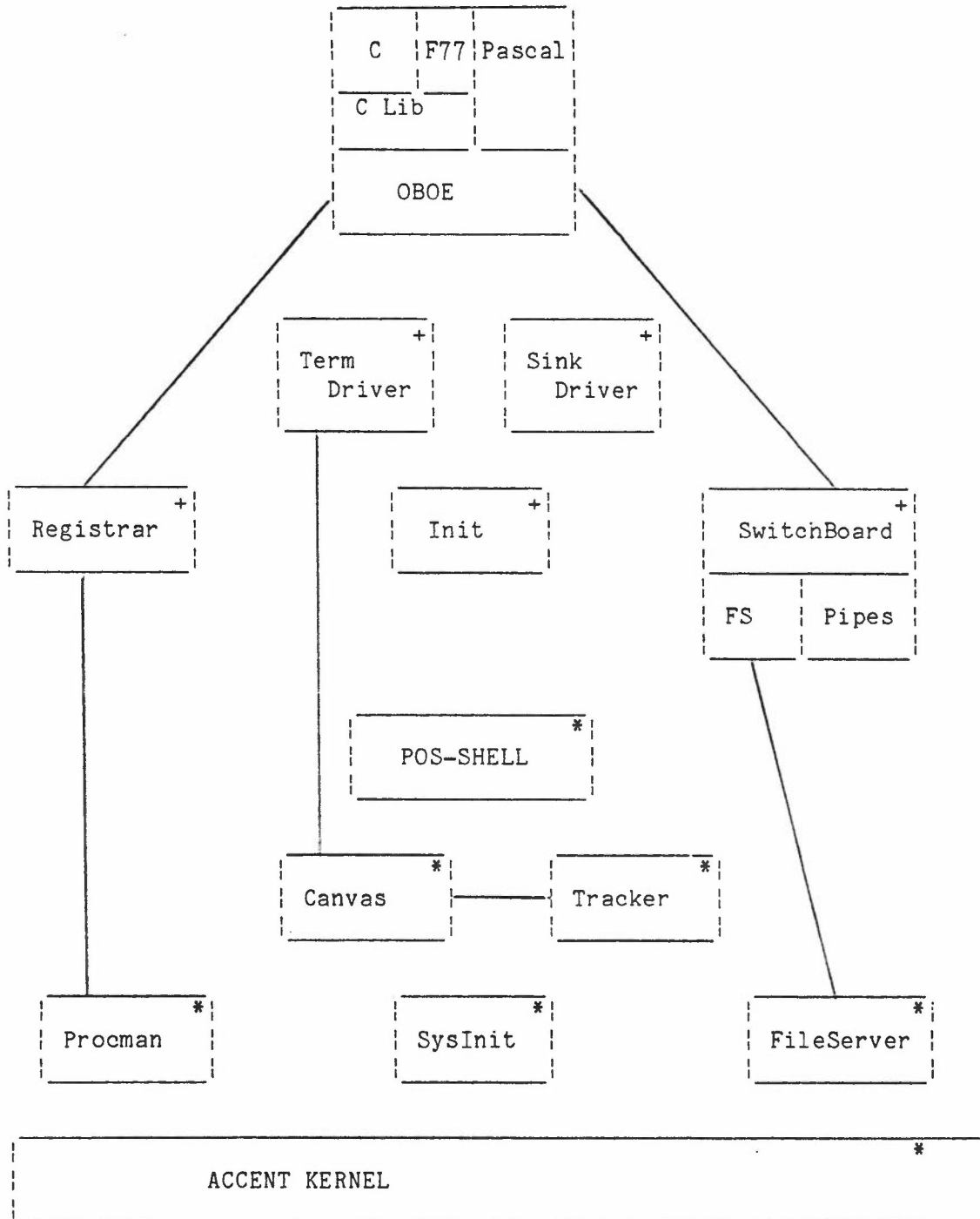Accent/Unix: Overall design   L.O.Ford

                                                  8 December 1982

    This paper attempts to give an overview of the underlying structure
of Accent/Unix. It is meant as guide to its process structure and as an
introduction to the various more detailed implementation papers.

    The Accent operating system kernel[14,15,13] on a Perq supports a
collection of processes. The kernel provides an execution environment
consisting of interprocess message communication (IPC), virtual memory
management and process management. All other services are provided by
processes that execute above the kernel.

    CMU have written processes which run above the Accent kernel to
provide a POS-like command environment. [2,3,18] Accent/Unix runs as a
normal user program in this environment. However, many extra processes
are spawned to support a Unix user process and the following diagram
shows the processes which support it.

Unix User Process

```
┌──────┬────┬────────┐
│  C   │F77 │ Pascal │
│      │    │        │
│──────┘    │        │
│ C Lib     │        │
│           │        │
│───────────┴────────│
│      OBOE          │
└────────────────────┘

┌──────────────+─┐   ┌──────────────+─┐
│  Term          │   │  Sink          │
│  Driver        │   │  Driver        │
│                │   │                │
└────────────────┘   └────────────────┘

┌────────────+─┐  ┌────────────+─┐  ┌──────────────────+─┐
│ Registrar    │  │    Init      │  │   SwitchBoard      │
│              │  │              │  │                    │
└──────────────┘  └──────────────┘  │────────┬───────────│
                                    │  FS    │  Pipes    │
                                    └────────┴───────────┘

        ┌────────────────────*─┐
        │                      │
        │   POS-SHELL          │
        └──────────────────────┘

┌──────────────*─┐    ┌──────────────*─┐
│  Canvas        │────│  Tracker       │
│                │    │                │
└────────────────┘    └────────────────┘

┌──────────────*─┐  ┌──────────────*─┐  ┌──────────────*─┐
│  Procman       │  │  SysInit       │  │  FileServer    │
│                │  │                │  │                │
└────────────────┘  └────────────────┘  └────────────────┘

┌──────────────────────────────────────────────────*─┐
│         ACCENT KERNEL                               │
└─────────────────────────────────────────────────────┘
```

* denotes process written at CMU to support the POS-like environment
+ denotes process written at RAL to support a Unix User process

Functions of various Accent/Unix Processes.

1. To Support the POS-like Shell (written by CMU)

1.1 FileServer

The Spice Interim Fileserver accepts requests from other processes to perform operations on file-structured devices. The Accent kernel does not impose any directory structure on the disk or floppy it only supports primitive disk operations such as ReadSegment, WriteSegment, CreateSegment and Destroy Segment. The functions of nameserver and fileserver are performed by the Spice Interim Fileserver. The Interim Fileserver supports those operations on the disk which were provided under POS and it is intended that in time this Interim Fileserver will be replaced by the final Spice Fileserver called Sesame[7].

1.2 Procman

Procman is the process manager for Accent/POS. It keeps track of all processes which are created and destroyed and makes them visible to the user. This allows the user to kill, suspend and resume, and debug any process. This process is also responsible for intercepting uncaught exceptions and calling the Mace debugger.

1.3 SysInit

SysInit performs system initialisation functions when the Accent/POS system is booted. It checks to see if the various servers, Canvas, Name and Network are present on the disk and if so starts them. (In Accent/Unix only the Canvas Server is started). It then starts the first POS-like shell and waits for it to die in which event it starts another.

1.4 Canvas

Canvas is the combined window and graphics manager. It accepts requests to create and destroy windows on the display and to perform graphical operations within those windows. [1]

1.5 Tracker

The tracker process handles the tablet and the keyboard and passes keys and cursor positions to Canvas.

2. Accent/Unix Processes

2.1 User Process

Accent/Unix supports almost a complete set of Unix V7 system calls. The interface seen by user code written in C or F77 is provided by routines in the C library and interfaces to OBOE which is written in POS-Pascal. POS-Pascal programs may call OBOE directly.

C library procedures provide name mapping and argument mapping between C and F77 programs and the POS-Pascal module OBOE (Off By One Error). This is necessary as some of the standard Unix system call

names are reserved words in POS-Pascal and also certain types such as strings need to be mapped between F77 C and POS-Pascal.

OBOE carries out any requests that it can perform directly and sends messages to the appropriate process to carry out those requests it cannot[6].

The routines which convert a procedure call into a message are generated automatically by Matchmaker[16]. Each User Process has a connection to Switchboard and Registrar established when the process is created by the Fork call and temporary connections are established to the other processes as necessary, i.e. on Open requests.

2.2 Switchboard

The Switchboard process conceptually consists of three separate processes whose functions have been combined into one process for efficiency[8,9]. These three processes are:

a)   Switchboard receives all open-like requests (open, stat, creat, etc) and determines which 'process', i.e. file manager, pipe manager or io driver the open is directed at. It passes on the request to the appropriate manager or driver. Subsequent read, write and close requests are sent directly to the appropriate 'process'.

b)   The File Manager handles all requests concerning the file system. On an open it looks up the name of the file by sending requests to the Spice Interim fileserver. If the file exists it maps the whole of the file into its virtual address space so that subsequent reads and writes to that file can be satisfied without going through the fileserver. Only when the file is closed by the last referencer is the file written back from the virtual address space of the File Manager to the disk, again using the fileserver to perform the function. The File Manager supports the concept of a file potentially having multi-readers and multi-writers which is not supported by the Spice Interim file server.

c)   The Pipe Manager supports Unix pipes by keeping queues of Accent messages[10]. Writes on a pipe cause messages to be queued on that pipe by the Pipe Manager and reads cause the messages to be dequeued. If the grain size of the read is different from that of the write then messages may have to be split or coalesced.

2.3 Registrar

Registrar is the process manager for Unix processes[4,5]. It keeps track of the births and deaths of Unix processes as well as their ancestry. Thus it makes visible to the user all Unix processes as procman makes visible all processes running on top of Accent, including Unix processes.

Registrar is also the Signal manager for Accent/Unix. It accepts signal requests from user or system processes and sends then, using Accent emergency messages, to the appropriate process or group of processes[17]. Alarm signals are also generated by Registrar.

Wait requests are handled by registrar and when it is notified that a child has died satisfies the wait request of the parent.

2.4 Init

Init is the father of all Unix processes[11]. It inherits all children whose parents have died. Its function during the Accent/Unix boot sequence is to start the two IO device drivers, Termdriver and Sinkdriver and to restart them on their death.

2.5 TermDriver

Termdriver is the process which handles requests to /dev/tty[12]. It performs keyboard input and display output by sending requests to Canvas. On receipt of a control C it sends a request to Registrar to send signals to all processes in its process group.

2.6 SinkDriver

Sinkdriver is the process which handles requests to /dev/null. It returns an EOF on a read and throws away all output.

Mappings of Unix system calls to processes

SWITCHBOARD

a) Switchboard

access	chmod	creat	ftime
link	mknod	mount	open
pipe	stat	stime	sync
time	umount	unlink	utime

b) File Manager / Pipe Manager / IO driver

close	fork	fstat	ioctl
lseek	read	write	

REGISTRAR

alarm	exit	fork	getgid
getegid	getpid	getuid	geteuid
kill	setgid	setegid	setuid
seteuid	signal	times	wait

OBOE

brk	sbrk	chdir	chroot
dup	dup2	exec	pause

Not implemented (NOPs)

acct	chown	lock	mpx
nice	phys	profil	ptrace

REFERENCES

1. J. EUGENE BALL, Canvas: the Spice graphics package, Dept of Computer
    Science, Carnegie-Mellon University, October 81.

2. ROGER B. DANNENBERG AND KATHLEEN FERRARO, Perq/Accent Introductory
    User Manual, Dept of Computer Science, Carnegie-Mellon University,
    November 82.

3. ROGER B. DANNENBERG AND KATHLEEN FERRARO, Perq/Accent Utilities
    Manual, Dept of Computer Science, Carnegie-Mellon University,
    November 82.

4. L.O. FORD, "Perq Unix Implementation Note # 42 - Accent/Unix: Regis-
    trar", DIC Note # 745, Rutherford Appleton Laboratory (December 82).
    [to be published].

5. L.O. FORD, "Perq Unix Implementation Note # 53 - Accent/Unix: Pro-
    cess ids and process groups", DIC Note # 758, Rutherford Appleton
    Laboratory (December 82). [to be published].

6. L.O. FORD, "Perq Unix Implementation Note # 48 - Accent/Unix: Exec
    System Call", DIC Note # 751, Rutherford Appleton Laboratory
    (December 82). [to be published].

7. MICHAEL B. JONES, RICHARD F. RASHID, AND MARY THOMPSON, Sesame: The
    Spice File System, Dept of Computer Science, Carnegie-Mellon Univer-
    sity, October 82.

8. J.C. MALONE, "Perq Unix Implementation Note # 41 - Accent/Unix:
    Switchboard", DIC Note # 744, Rutherford Appleton Laboratory
    (December 82). [to be published].

9. J.C. MALONE, "Perq Unix Implementation Note # 51 - Accent/Unix: Spe-
    cial files, mountable devices", DIC Note # 756, Rutherford Appleton
    Laboratory (December 82). [to be published].

10. J.C. MALONE, "Perq Unix Implementation Note # 52 - Accent/Unix:
    Pipes", DIC Note # 757, Rutherford Appleton Laboratory (December
    82). [to be published].

11. J.C. MALONE, "Perq Unix Implementation Note # 43 - Accent/Unix: Boot
    Sequence", DIC Note # 746, Rutherford Appleton Laboratory (December
    82). [to be published].

12. J.C. MALONE, "Perq Unix Implementation Note # 44 - Accent/Unix:
    TermDriver", DIC Note # 747, Rutherford Appleton Laboratory
    (December 82). [to be published].

13. R.F. RASHID, Accent Kernel Interface Manual, Dept of Computer Sci-
    ence, Carnegie Mellon University, January 1982.

14. RICHARD F. RASHID AND GEORGE G. ROBERTSON, "Accent: A communication
    oriented network operating system kernel", ACM, 64-74 (December 81).

15. RICHARD F. RASHID AND GEORGE G. ROBERTSON, Accent: A communication oriented network operating system kernel, Dept of Computer Science, Carnegie-Mellon University, April 81.

16. MARY THOMPSON, MICHAEL B. JONES, KATHIE FERRARO, AND KEITH WRIGHT, Matchmaker: A Remote Procedure Call Generator, Dept of Computer Science, Carnegie-Mellon University, November 82.

17. A.S. WILLIAMS, "Perq Unix Implementation Note # 50 - Accent/Unix: Signals and deadports", DIC Note # 755, Rutherford Appleton Laboratory (December 82). [to be published].

18. KEITH WRIGHT, Oil: The Spice Ascii Editor, Dept of Computer Science, Carnegie-Mellon University, April 82.