# Operating systems

**Richard Harvey** 





# Operating systems

### **Richard Harvey**

IT Livery Company Professor of Information Technology, Gresham College

# GRESHAM COLLEGE

www.prof-richard.org



## Before Operating Systems...



"Programmers at the console of a UNIVAC 1, National Museum of American History, Behring Center, https://amhistory.si.edu/img/cobol/COBOL002-800.jpg





### **OS Timeline**



Mac OS				
S-DOS Windows	Palm OS		Web O IOS Android	S WatchOS
Sun OS	Linux	Novell Netware		
ر م و ر	9			



### "Ideas of fundamental interest"

Table 2 Fundamental Ideas			
	Major Phases	Technical Innovations	
1	Open Shop	The idea of operating syste	
11	Batch Processing	Tape batching	
		First-in, first-out schedulin	
III	Multiprogramming	Processor multiplexing	
		Indivisible operations	
		Demand paging	
		Input/output spooling	
		Priority scheduling	
		Remote job entry	
IV	Timesharing	Simultaneous user interact	
		On-line file systems	
V	<b>Concurrent</b> Programming	Hierarchical systems	
		Extensible kernels	
		Parallel programming conc	
		Secure parallel languages	
Vl	Personal Computing	Graphic user interfaces	
V11	Distributed Systems	Remote servers	



## Slow peripherals



HP 2631G Dot Matrix Printer Restoration (vintage 1978)

https://www.youtube.com/watch?v=54bu0bM0Qjo

### The UNIX Time-Sharing System

Dennis M. Ritchie and Ken Thompson Bell Laboratories

UNIX is a general-purpose, multi-user, interactive operating system for the Digital Equipment Corporation PDP-11/40 and 11/45 computers. It offers a number of features seldom found even in larger operating systems, including: (1) a hierarchical file system incorporating demountable volumes; (2) compatible file, device, and inter-process I/O; (3) the ability to initiate asynchronous processes; (4) system command language selectable on a per-user basis; and (5) over 100 subsystems including a dozen languages. This paper discusses the nature and implementation of the file system and of the user command interface.

Key Words and Phrases: time-sharing, operating system, file system, command language, PDP-11

CR Categories: 4.30, 4.32

#### 1. Introduction

There have been three versions of UNIX. The earliest version (circa 1969–70) ran on the Digital Equipment Corporation PDP-7 and -9 computers. The second version ran on the unprotected PDP-11/20 computer. This paper describes only the PDP-11/40 and /45 [1] system since it is more modern and many of the differences between it and older UNIX systems result from redesign of features found to be deficient or lacking.

Since PDP-11 UNIX became operational in February 1971, about 40 installations have been put into service; they are generally smaller than the system described here. Most of them are engaged in applications such as the preparation and formatting of patent applications and other textual material, the collection and processing of trouble data from various switching machines within the Bell System, and recording and checking telephone service orders. Our own installation is used mainly for research in operating systems, languages, computer networks, and other topics in computer science, and also for document preparation.

Perhaps the most important achievement of UNIX is to demonstrate that a powerful operating system for interactive use need not be expensive either in equipment or in human effort: UNIX can run on hardware costing as little as \$40,000, and less than two manyears were spent on the main system software. Yet UNIX contains a number of features seldom offered even in much larger systems. It is hoped, however, the users of UNIX will find that the most important characteristics of the system are its simplicity, elegance, and ease of use.

Besides the system proper, the major programs available under UNIX are: assembler, text editor based on QED [2], linking loader, symbolic debugger, compiler for a language resembling BCPL [3] with types and structures (C), interpreter for a dialect of BASIC, text formatting program, Fortran compiler, Snobol interpreter, top-down compiler-compiler (TMG) [4], bottom-up compiler-compiler (YACC), form letter generator, macro processor (M6) [5], and permuted index program.

There is also a host of maintenance, utility, recreation, and novelty programs. All of these programs were written locally. It is worth noting that the system is totally self-supporting. All UNIX software is maintained under UNIX; likewise, UNIX documents are generated and formatted by the UNIX editor and text formatting program.

#### 2. Hardware and Software Environment

The PDP-11/45 on which our UNIX installation is implemented is a 16-bit word (8-bit byte) computer with 144K bytes of core memory; UNIX occupies 42K bytes. This system, however, includes a very large number of device drivers and enjoys a generous allotment of space for I/O buffers and system tables; a minimal system

Communications of the ACM July 1974 Volume 17 Number 7



### Ken Thompson and Dennis Ritchie explain Unix (Bell Labs)

https://www.youtube.com/watch?v=JoVQTPbD6UY



Copyright © 1974, Association for Computing Machinery, Inc. General permission to republish, but not for profit, all or part of this material is granted provided that ACM's copyright notice is given and that reference is made to the publication, to its date of issue, and to the fact that reprinting privileges were granted by permission of the Association for Computing Machinery.

This is a revised version of a paper presented at the Fourth ACM Symposium on Operating Systems Principles, IBM Thomas J. Watson Research Center, Yorktown Heights, New York, October 15–17, 1973. Authors' address: Bell Laboratories, Murray Hill, NJ 07974.



user

### kernel

### hardware



### Linux

#### Linux bcc/BPF Tracing Tools





https://github.com/iovisor/bcc#tools 2019



## Virtual memory - demand paging



Speak roughly to your little VAX, And boot it when it crashes; It knows that one cannot relax Because the paging thrashes! Wow! Wow! Wow!

I speak severely to my VAX, And boot it when it crashes; In spite of all my favorite hacks My jobs it always thrashes! Wow! Wow!

### Monolithic versus micro-kernels





### Libraries







## Solaris scheduling

	% dispadmin - c TS -g					
	<pre># Time Sharing Dispatcher Configura</pre>		uration			
	RES=1000					
	# ts_quantum	ts_tqexp	ts_slpret	ts_maxwait	PRIO	RITY LEVE
Thread Class	200	0	0	50	#	0
Time charing (TC)	200	0	0	50	#	1
rime sharing (15)	200	0	0	50	# #	2
Intoractive (IA)	200	0	0	50	# #	5 Д
Interactive (IA)	200	0	0	50	#	5
Poaltimo (PT)	200	0	0	50	#	6
	200	0	0	50	#	7
System (SVS)						
Jystern (J1J)	40	38	0	58	#	48
Fair share (FSS)	40	39	0	59 50	# #	49 50
	40	40 41	0	59	# #	51
Fixed priority (FP	40	42	0	59	#	52
	40	43	0	59	#	53
	40	44	0	59	#	54
	40	45	0	59	#	55
	40	46	0	59	#	56
	40	4./	0	59	#	57
	40	48 19	32000	59 59	#	58 59
		4 <i>3</i>	52000		11	33

IPC Expected X=0X=0sum() { read x write x = x + 1} x=1sum() { read x write x = x + 1 } x=2



x=1

## Priority inversion

![](_page_17_Picture_1.jpeg)

From <u>https://mars.nasa.gov/resources/8529/sojourner-yogi/</u>

### **The cloud**

![](_page_18_Figure_1.jpeg)

Adapted from laaS versus PasS versus SaaS, IBM Learn, <u>https://www.ibm.com/cloud/learn/iaas-paas-saas</u>

![](_page_18_Figure_5.jpeg)

### Virtualisation - laaS

### processes

kernel

hardware

processes	processes	processes		
kernel	kernel	kernel		
VM 1	VM 2	VM 3		
VM manager				
hardware				

### Market Share of Operating Systems for Personal Devices (Desktop, Workstation, Tablet and Smartphone)

![](_page_20_Figure_1.jpeg)

by A. Eylenburg, eylenburg.github.io

### Next season

### A new Gresham Professor and a new series

### Watch the website for details

Thanks and kudos to the Worshipful Company of Information Technologists who sponsor these lectures.