

The Art of Unix Programming

by Eric Steven Raymond

The Art of Unix Programming

by Eric Steven Raymond

Copyright © 2003 Eric S. Raymond

This book and its on-line version are distributed under the terms of the Creative Commons Attribution-NoDerivs 1.0 license, with the additional proviso that the right to publish it on paper for sale or other for-profit use is reserved to Pearson Education, Inc. A reference copy of this license may be found at <http://creativecommons.org/licenses/by-nd/1.0/legalcode>.

AIX, AS/400, DB/2, OS/2, System/360, MVS, VM/CMS, and IBM PC are trademarks of IBM. Alpha, DEC, VAX, HP-UX, PDP, TOPS-10, TOPS-20, VMS, and VT-100 are trademarks of Compaq. Amiga and AmigaOS are trademarks of Amiga, Inc. Apple, Macintosh, MacOS, Newton, OpenDoc, and OpenStep are trademarks of Apple Computers, Inc. ClearCase is a trademark of Rational Software, Inc. Ethernet is a trademark of 3COM, Inc. Excel, MS-DOS, Microsoft Windows and PowerPoint are trademarks of Microsoft, Inc. Java. J2EE, JavaScript, NeWS, and Solaris are trademarks of Sun Microsystems. SPARC is a trademark of SPARC international. Informix is a trademark of Informix software. Itanium is a trademark of Intel. Linux is a trademark of Linus Torvalds. Netscape is a trademark of AOL. PDF and PostScript are trademarks of Adobe, Inc. UNIX is a trademark of The Open Group.

The photograph of Ken and Dennis in Chapter 2 appears courtesy of Bell Labs/Lucent Technologies.

The epigraph on the Portability chapter is from the Bell System Technical Journal, v57 #6 part 2 (July-Aug. 1978) pp. 2021-2048 and is reproduced with the permission of Bell Labs/Lucent Technologies.

Dedication

To Ken Thompson and Dennis Ritchie, because you inspired me.

Table of Contents

Preface	xvi
Who Should Read This Book	xvii
How to Use This Book	xviii
Related References	xix
Conventions Used in This Book	xx
Our Case Studies	xxi
Author's Acknowledgements	xxii
I. Context	24
1. Philosophy	25
Culture? What Culture?	25
The Durability of Unix	25
The Case against Learning Unix Culture	27
What Unix Gets Wrong	28
What Unix Gets Right	29
Open-Source Software	29
Cross-Platform Portability and Open Standards	29
The Internet and the World Wide Web	30
The Open-Source Community	30
Flexibility All the Way Down	31
Unix Is Fun to Hack	32
The Lessons of Unix Can Be Applied Elsewhere	32
Basics of the Unix Philosophy	33
Rule of Modularity: Write simple parts connected by clean interfaces.	36
Rule of Clarity: Clarity is better than cleverness.	36
Rule of Composition: Design programs to be connected with other programs.	37
Rule of Separation: Separate policy from mechanism; separate interfaces from engines.	38
Rule of Simplicity: Design for simplicity; add complexity only where you must.	39
Rule of Parsimony: Write a big program only when it is clear by demonstration that nothing else will do.	40
Rule of Transparency: Design for visibility to make inspection and debugging easier.	40

Rule of Robustness: Robustness is the child of transparency and simplicity.	41
Rule of Representation: Fold knowledge into data, so program logic can be stupid and robust.	42
Rule of Least Surprise: In interface design, always do the least surprising thing.	42
Rule of Silence: When a program has nothing surprising to say, it should say nothing.	43
Rule of Repair: Repair what you can — but when you must fail, fail noisily and as soon as possible.	44
Rule of Economy: Programmer time is expensive; conserve it in preference to machine time.	45
Rule of Generation: Avoid hand-hacking; write programs to write programs when you can.	45
Rule of Optimization: Prototype before polishing. Get it working before you optimize it.	46
Rule of Diversity: Distrust all claims for one true way.	47
Rule of Extensibility: Design for the future, because it will be here sooner than you think.	48
The Unix Philosophy in One Lesson	48
Applying the Unix Philosophy	49
Attitude Matters Too	50
2. History	52
Origins and History of Unix, 1969-1995	52
Genesis: 1969–1971	52
Exodus: 1971–1980	55
TCP/IP and the Unix Wars: 1980-1990	57
Blows against the Empire: 1991-1995	64
Origins and History of the Hackers, 1961-1995	66
At Play in the Groves of Academe: 1961-1980	67
Internet Fusion and the Free Software Movement: 1981-1991	68
Linux and the Pragmatist Reaction: 1991-1998	71
The Open-Source Movement: 1998 and Onward	73
The Lessons of Unix History	75
3. Contrasts	76
The Elements of Operating-System Style	76
What Is the Operating System’s Unifying Idea?	76

[Click here to download full PDF material](#)