A Brief History of Unix

Tom Ryder
tom@sanctum.geek.nz
https://sanctum.geek.nz/

I Love Unix ∴ I Love Linux

- When I started using Linux, I was impressed because of the **ethics** behind it.
- I loved the idea that an operating system could be both free to customise, and free of charge.
  - Being a cash-strapped student helped a lot, too.
- As my experience grew, I came to appreciate the **design** behind it.
- And the design is **UNIX**.
- Linux isn’t a perfect Unix, but it has all the really important bits.

What do we actually mean?

- We’re referring to the **Unix family** of operating systems.
  - Unix from Bell Labs (Research Unix)
  - **GNU/Linux**
  - Berkeley Software Distribution (BSD) Unix
  - Mac OS X
  - Minix (Intel loves it)
  - ...and many more

Warning signs: 1/2

If your operating system shows many of the following symptoms, it may be a Unix:

- Multi-user, multi-tasking
- Hierarchical filesystem, with a single root
- Devices represented as files
- Streams of text everywhere as a user interface
- “Formatless” files
  - Data is just data: streams of bytes saved in sequence
  - There isn’t a “text file” attribute, for example

Warning signs: 2/2

- Bourne-style shell with a “pipe”:
  - `$ program1 | program2`
- “Shebangs” specifying file interpreters:
  - `#!/bin/sh`
- C programming language baked in everywhere
- Classic programs: `sh(1)`, `awk(1)`, `grep(1)`, `sed(1)`
- Users with beards, long hair, glasses, and very strong opinions...

Nobody saw it coming!

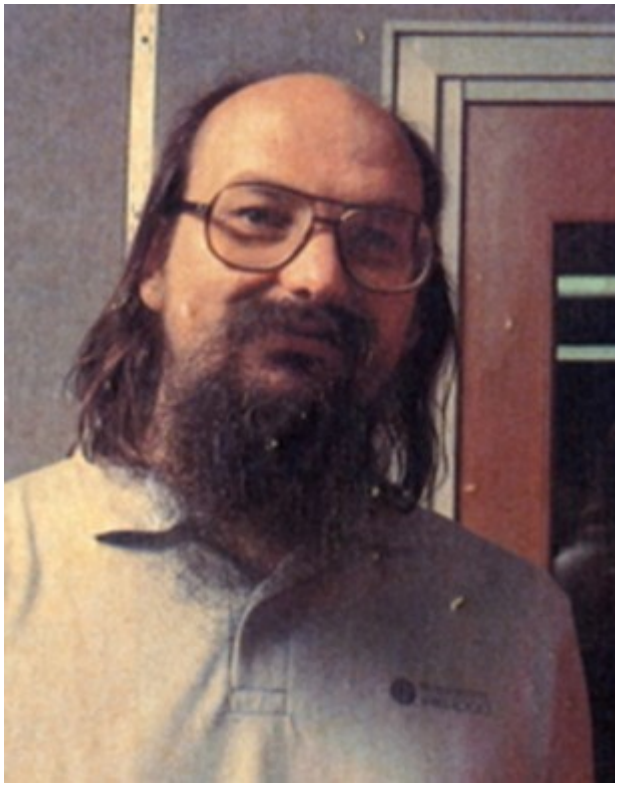
> “The number of Unix installations has grown to 10, with more expected.”
>
> — Ken Thompson and Dennis Ritchie (1972)

- Unix in some flavour is in servers, desktops, embedded software (including Intel’s management engine), mobile phones, network equipment, single-board computers...
- If Android counts as a Unix (arguable), then Unix is the most widely-deployed operating system design **ever**.
  - It’s not a fair fight. It’s not even close.

Genesis

- Where did Unix come from?
- What made it so different?
- What made it so successful?

Our Founder



Kenneth Lane Thompson
/usr/ken

Err, Our Founders, Rather

Kenneth Lane Thompson and Dennis MacAlistair Ritchie
`/usr/ken` and `/usr/dmr`

Bell Labs and Multics

- Ken was employed at Bell Labs straight out of his Master's at Berkeley.
- He'd been employed to work on operating systems, which had an uncertain future at the labs.
- Bell Labs was a government monopoly, and dedicated a significant portion of their massive budget to research, including operating systems.
- Ken and Dennis were both working on the **Multics** operating system project.

The End of Multics

- Multics was supposed to *hide* the complexities of software from the users and make computing easier to manage.
- Instead, it became bloated and unwieldy itself, and entered **development hell**.
- Eventually Bell Labs canned the project.
- And Ken got bored...
  - (He also missed having a computer to run his “Space Travel” game on)

The Little PDP-7 That Could

- On a “scavenged” PDP-7, Ken began working on what started as just a file system.
- It grew an operating system by accident:

  Seibel: So you basically wrote an OS so you'd have a better environment to test your file system.

  Thompson: Yes. Halfway through there that I realized it was a real time-sharing system.
- Within a few months (!), with support from other researchers, Ken had an **assembler**, an **editor**, and perhaps most revolutionary, a new kind of **command-line shell**.

ALGOL, BCPL, B, C

- A programming language was needed for the new system.
- The **ALGOL**-inspired language **BCPL** was popular, and had much of what Ken needed...
  - ...and a lot more of what he didn’t.
- It had to be made smaller to work reliably on Unix on the PDP-7
- He made a stripped-down, typeless version called **B.**
- Dennis “took [B] and added data types”, and thus was born “New B”, and then, more logically, **C.**
- Unix and C have been inseparable from the beginning.

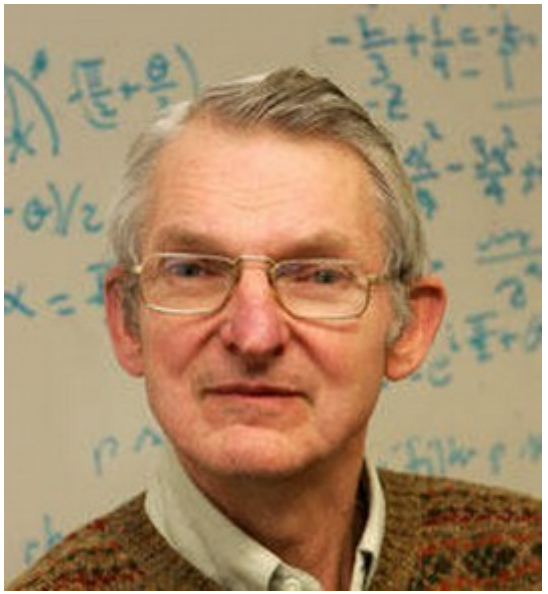
Work ain’t done ‘til the paperwork is

- There are quite a few tools in the original Unix relevant to document preparation:
  - Text processing tools (`grep`, `tr`, `sed`, `awk`)
  - Typesetting (`roff`, `man`)
- There are two reasons for this:

  1) The creators were themselves interested in the topics, and believed strongly in good documentation.

  2) It was one of the ways they justified continuing development on the system to management, as such systems were needed internally at Bell Labs.
- This turned out to be another reason Unix did so well: it was *documented*, and the documentation was *good* and *accessible*.

Ceci n’est pas une pipe.



- **Doug McIlroy** is a relatively unsung Unix hero.
- **Pipes** were essentially his idea.
- When people are asked what makes the Unix shell different, their answer will usually involve a **pipeline**:

```
$ tr -cs A-Za-z '\n' | tr A-Z a-z | sort |
uniq -c | sort -rn | sed ${1}q
```

AWK and K&R

- **Brian Kernighan** developed AWK with **Al Aho** and **Peter Weinberger**.
- Convenient way to use much of C's power in pipelines without having to compile a program.
- Still useful today—and highly underrated!
- Also wrote "The C Programming Language" with Ritchie...known as "K&R".

Bootstrapping Unix

- The conventional wisdom was that all operating systems had to be coded in the host system's **assembly language**.
- Unix was **rewritten in C** so that the whole operating system, including its kernel and development tools, could be **ported**.
- It wasn't until later on that the implications of this became obvious:

  Operating systems were no longer bound to specific hardware!
- This was a huge reason that Unix won despite the marketing clout of giants like IBM, and it ended up reshaping the market.

Legal grey area

- Outside interest in the operating system grew to be considerable, especially from academia.
  - A multi-user operating system built by computer scientists, for computer scientists, that runs on a wide variety of hardware, and doesn't require a fat contract to IBM?
  - Hobbyists and universities asked: Where do we sign?
- But Bell Labs was a **regulated monopoly**.
  - They couldn't sell Unix—that would be anticompetitive.
- The solution?

With love from Bell Labs

Thompson quietly mailed out tapes and disks with the complete system to interested parties, signing each one:

“Love, ken”

Social OS: 1/2

- This makes more sense when you look at some of the attitudes that drove the Unix creators:

  “We have not been faced with the needs to satisfy someone else's requirements, and **for this freedom we are thankful**.”
  —Thompson and Ritchie

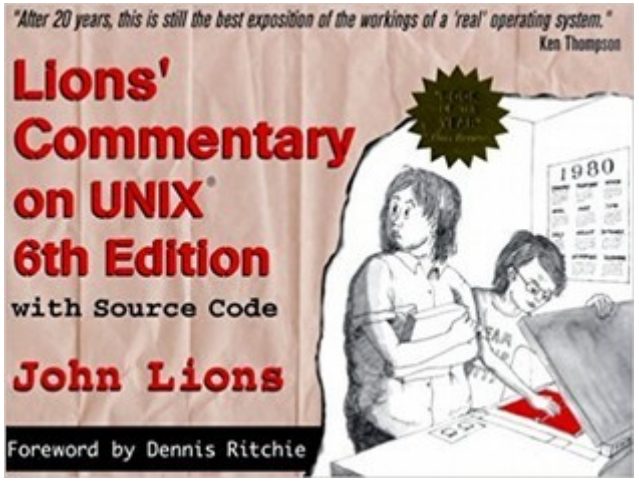
  “What we wanted to preserve was not just a good environment in which to do programming, but a **system around which a fellowship could form**.”
  —Ritchie

Social OS: 2/2

- A great deal of Unix's design isn't so much **user-friendly** as it is **users-friendly**:
  - Multiple concurrent users on then-limited hardware
  - Plain text interfaces and scripts
  - Open filesystem design
- "The Unix room still exists, and it may be the greatest cultural reason for the success of Unix as a technology."
  — Rob Pike
- If you've ever been lucky enough to share a networked Unix-like system with a few other like-minded people in the same room, you know how much fun this can be.

“Hey, kid…wanna buy some Unix?”

- The growing interest in Unix was largely due to universities in the U.S. adopting it for teaching.
- The complete source code to Unix v6 was published in “The Lions Book”—a bit of legally dubious *samizdat*.

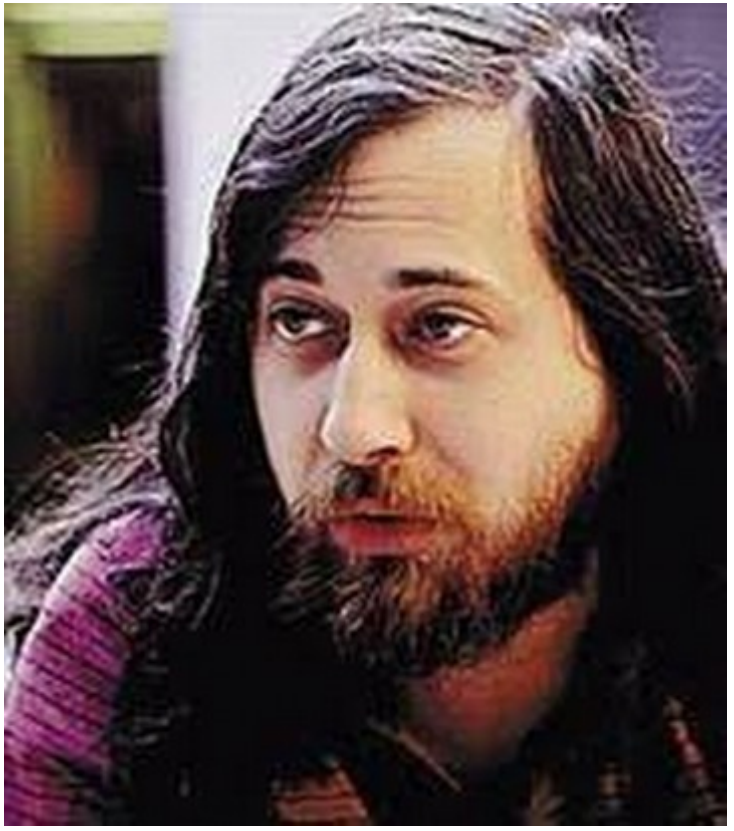


Despite having to be clandestinely copied for much of its history, it was an explosively popular publication in universities.

`cat(1)` came back from Berkeley waving flags

- With access to the source code, universities with the manpower (and student interest) would customise or extend Unix to suit their needs.
- This led to a slowly more and more divergent Unix ecosystem.
- **Berkeley Software Distribution (BSD) Unix** is the true survivor here, living on in FreeBSD, OpenBSD, and NetBSD.
- Lots of extensions to Unix:
  - TCP/IP networking
  - Bill Joy's `vi(1)` editor, which we now use mostly as **Vim**
  - Bizarre new switches for `cat(1)`...

The last hacker (nearly)



- Meanwhile, LISP hacker **Richard Stallman**'s beloved AI lab at MIT had finally fallen apart.
- He attributed this to the malicious influence of **proprietary software**, undermining the open, shared, collaborative values.
- He wanted a **free operating system**, to keep that spirit alive.

So close, yet so far

- Much of the original collaborative spirit of Unix was (and now is) in BSD Unix.
- But early attempts at selling it commercially attracted the attention of **AT&T**, the parent company of Bell Labs.
- A lawsuit, settled out of court, effectively shut it down in 1992: *UNIX System Laboratories, Inc. v. Berkeley Software Design, Inc.*
- It became clear that BSD would not be the free Unix that Stallman and other users wanted.

Brave GNU World

- Stallman's **GNU's not Unix (GNU) Project** was announced in 1983, and interested hackers got to work replacing proprietary software:
  - `bc, dc, grep, sed`…
  - GNU Bash
  - And perhaps most importantly: The **GNU Compiler Collection (GCC)**
- Stallman was not fond of Unix, but considered imitating it to be the best shot he had at accomplishing his goals.

The Hollow OS

- By the early 90s, progress had stalled.
- GNU tools were being more and more widely adopted, but on **proprietary systems**.
- This was better than nothing, but not what Stallman had envisioned. He wanted a **complete system**.
- But the **GNU HURD**, the kernel, was stuck in **development hell**...

“just a hobby, won't be big and professional like gnu”

- **Linus Torvalds** was a student at the University of Helsinki in Finland.
- He wanted a Unix-like operating system that ran on the 386.
- He had Andrew Tanenbaum’s academic tool OS, **MINIX**, to work from and imitate.
- So he wrote a kernel, and got GNU software (GCC and Bash) running on it...

Right place, right `time(2)`

- Fortunately, around that time, Linus had been to hear Richard Stallman speak about free software and the **GNU Public License**.
- He liked what he heard, though he and Stallman do not agree on many details.
- He decided to release his new toy project under that license...

...and here we all are today, with the spirit of Unix still everywhere.

Questions?

- **This topic runs *deep*.** I've left out a lot of really important people and developments.
- The history of Unix is **well-documented** by many people, and deeply fascinating.
  - Wikipedia's article is decent: https://en.wikipedia.org/wiki/History_of_Unix
  - Eric Raymond's account is also great reading (and openly opinionated): http://www.catb.org/esr/writings/taoup/html/historychapter.html
  - cat-v.org has some carefully-archived Unix historical documents, including ancient manual pages: http://doc.cat-v.org/unix/

- Email: tom@sanctum.geek.nz
- Website: https://sanctum.geek.nz/