

# The Chronicles of GNU/Linux



**Tom Ryder**

[tom@sanctum.geek.nz](mailto:tom@sanctum.geek.nz)  
<https://sanctum.geek.nz/>

# The saga so far... (1/3)

- **Unix** is being refined, and **hackers love it!**
  - It's **simple!**
  - It's **portable!**
  - It's **documented!** (*never* underestimate this one)
  - It has **C!**
  - It's free for ev ... oh wait.
- Well, this sucks. Maybe we can get a free Unix again somehow?

# The saga so far... (2/3)

- **BSD**, based on University of California's extensive modifications to Unix, initially looks hopeful, but gets slapped down with a lawsuit from AT&T.
- Looks like Ma Bell isn't letting her baby go without a fight...

# The saga so far... (3/3)

- Richard Stallman's **GNU (GNU's Not Unix) system** is designed to be a **free (libre) Unix**, and its tools are already proving popular on other systems:
  - GNU libc
  - GNU Compiler Collection
  - GNU Bourne-Again Shell
- It's looking more and more like a **complete system**, the more the Free Software Foundation work on it.

# So close, and yet so far

- There are a few things every **self-hosted, self-respecting Unix** is going to need.
- Stallman started by himself, with the software he'd already written (Emacs), and recruited others to *completely remake* all of these things.
- By 1990, the GNU project had *all of them...* except one.
  - This single “missing piece” shaped computing history.

# Free Unix checklist (1/6)

- You're gonna need a **text editor**:
  - To write programs and documents on the system.
  - To configure the system itself.

✓ **Done! (1985):** GNU Emacs



(If you didn't like Emacs, you were a bit stuck...  
Bram Moolenaar wouldn't make Vim until 1991.)

# Free Unix checklist (2/6)

- You're gonna need a **compiler** for the **C programming language**:
  - To translate C statements like `i++` to assembly like `incw %ax`, and then into **binary objects** and **programs**.
  - To preprocess, compile, assemble, and link.

✓ **Done! (1987):** The GNU C Compiler



(Huge deal all by itself: suddenly, you could code in C without having to buy a proprietary compiler!)

# Free Unix checklist (3/6)

- You're gonna need an implementation of the **standard library for C**:
  - Defining crucial functions like `malloc(3)` and `printf(3)`.

✓ **Done! (1987): glibc**

(Works great with GCC!)



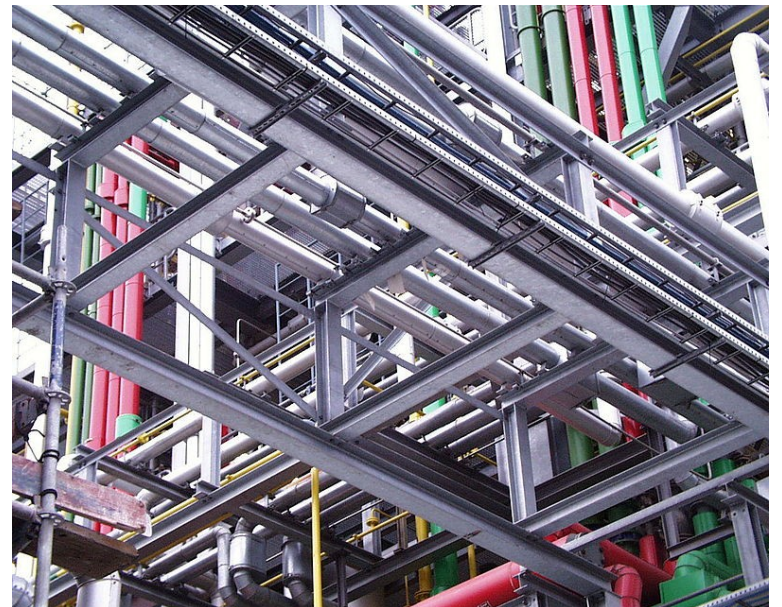


# Free Unix checklist (4/6)

- You're gonna need the rest of the **Unix userland tools**:
  - ls, bc, dc, diff, grep, comm, awk, cmp, sed, sort, tr, more, m4, rm, cp, ln, pr, fmt, ed, mv...

✓ **Done!** GNU coreutils (fileutils, shellutils, textutils), GNU grep, gawk, gsed...

*Whew!* This **reimplementing the entirety of Unix** thing is a lot of work...



# Free Unix checklist (5/6)

- You're gonna need **a command-line shell**:
  - To allow users an interface to the system.
  - To use all those fancy Unix tools you just implemented.

```
$ grep foo < in | sort -k1,2 | uniq > out
```

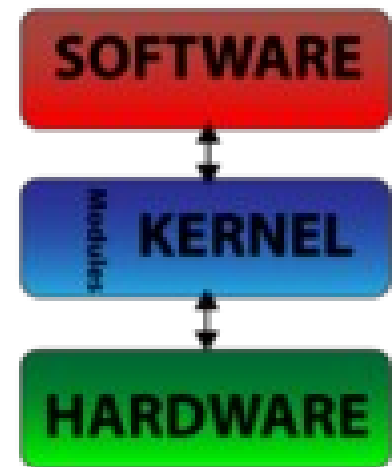
✓ **Done! (1989):** GNU Bourne-Again Shell (bash)

(Yes, Bash predates Linux!)



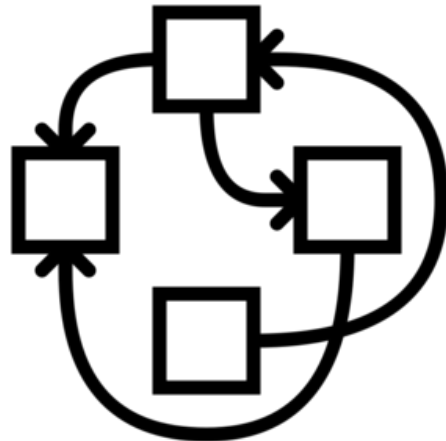
# Free Unix checklist (6/6)

- You're gonna need a **kernel**:
  - To translate **system calls** from programs into actual **activity** on the computer hardware.
  - To manage the system's **memory** and other resources.
- GNU's kernel is called the **GNU/HURD**, with development begun in 1990.
- And its first stable release was in...
  - ...umm
  - ...uhh



# Nobody is immune

- The **GNU/HURD** kernel has been in **development hell** for **nearly 30 years**.
- There are usable experimental implementations, but it's still not finished.
- And so the GNU system remained incomplete, until...



# “Just a hobby”

From: torv...@klaava.Helsinki.FI (Linus Benedict  
Torvalds)  
Newsgroups: comp.os.minix  
Subject: What would you like to see most in minix?  
Date: 25 Aug 91 20:57:08 GMT  
Organization: University of Helsinki

Hello everybody out there using minix -

I'm doing a (free) operating system (just a hobby, won't be big and professional like gnu) for 386(486) AT clones. This has been brewing since april, and is starting to get ready. I'd like any feedback on things people like/dislike in minix, as my OS resembles it somewhat...

# The mother of invention

- Linus Torvalds hacked on Andrew Tanenbaum's **MINIX**, a 16-bit operating system used as an educational tool.
- He was frustrated by the absence of a free kernel for the 386, and moved to fill that void.
  - If 386BSD or the GNU/HURD had been workable, he says he would have used them.
- Like any good hacker, finding nothing to fit his needs, he **made his own**.

# An early entangling

GNU and Linux have been intertwined right from the beginning.

- The kernel was developed on MINIX, but compiled using the **GNU C Compiler**.
- Linus saw fit to make **GNU Bash** and the **GNU C Compiler** work on his new system as a priority.
- Early releases of Linux **bundled** GNU tools.
- It was a perfect fit *technically*, right from the start...

# Trouble in paradise

...but not *ideologically*.

- For **Torvalds**, freedom was a means to *practical* ends: better software, a better community, and less artificial roadblocks to development. Linux was **Open Source**.
- For **Stallman**, freedom was the *goal in itself*: the source being open for anyone to read was just a necessary precondition, not the point in itself. GNU was **Free Software**.



# What's in a name? (1/3)

- People couldn't even agree on what the complete new system was *called*.
- Most people called it just “Linux”, much to Stallman's despair:
  - What about all that effort the Free Software Foundation went to?
  - Why was the whole operating system named after the kernel?
  - What if people missed the ideology and ideas behind GNU?

# What's in a name? (2/3)

- Richard Stallman and the Free Software Foundation would like you to call it **GNU/Linux**, or **GNU+Linux**.
- **Debian GNU/Linux** agrees. So does your presenter, but that battle has been lost, because...
- **Red Hat Linux, SUSE Linux, Arch Linux, Void Linux**, and most other Linux distributions and vendors *don't*.

# What's in a name? (3/3)

- Linus himself conceded this gap very early on, in the release notes for Linux v0.1:

**“Sadly, a kernel by itself gets you nowhere. To get a working system you need a shell, compilers, a library etc. These are separate parts and may be under a stricter (or even looser) copyright. Most of the tools used with linux are GNU software and are under the GNU copyleft.”**

# 1991: First release

- A simple “not for commercial distribution” license, uploaded to Finnish university servers.
- Initially called “Freax”, but the administrator renamed the directory to “Linux”.
  - Linus thought that name was too egotistical at first.
  - Yes, you read that right.

# 1992: GPL release

- The first few versions released under the **GNU Public License** are created.
- The GPL preserves the **Four Freedoms**:
  - **Freedom 0**: Run the program in any way for any reason.
  - **Freedom 1**: Study and modify the program.
  - **Freedom 2**: Share copies of the program.
  - **Freedom 3**: Share *modifications* to the program.
- Linus later said that making the kernel GPL was “one of the best things I ever did”.

# 1993: Development and distribution

- Only two years into the kernel's existence, **more than 100 developers** work on it during that year.
- **Slackware Linux** is released...the oldest distribution still active today.
- **Debian GNU/Linux** is released...the largest “community” distribution today.



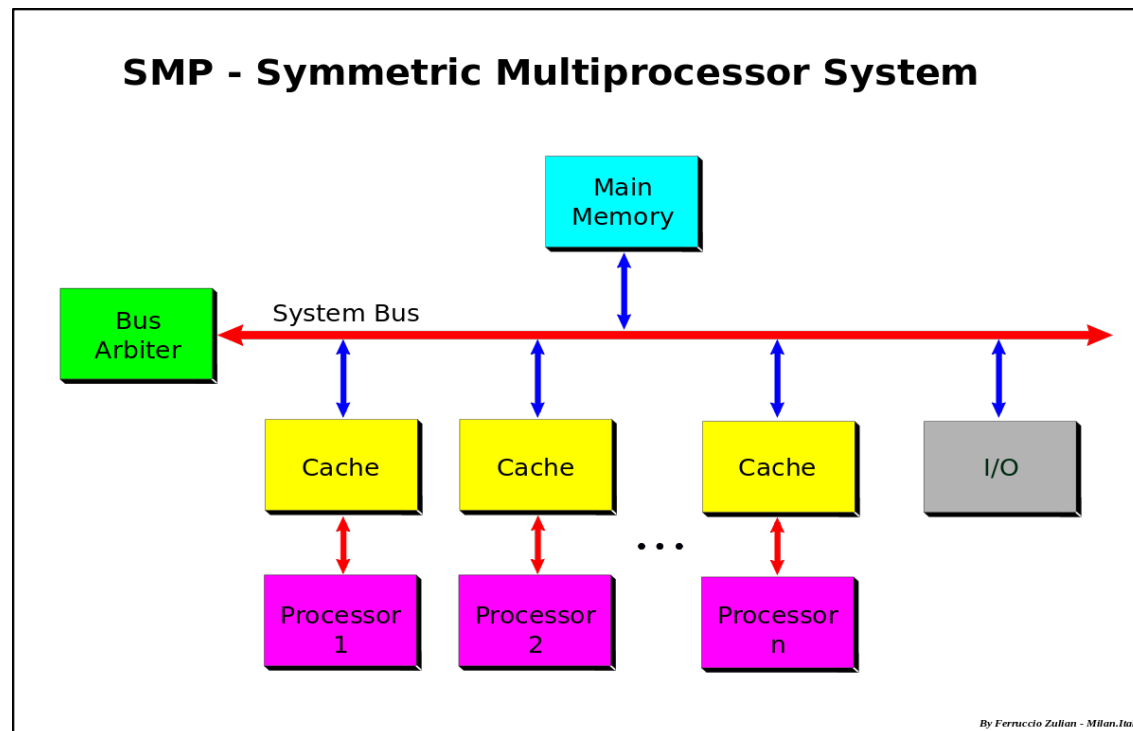
# 1994: Version 1.0

- All components of the kernel are now “mature”, in Linus’ reckoning.
- **Red Hat Linux** and **SUSE Linux** make their first public releases; commercially supported Linux becomes a reality in the following few years.



# 1996 (1/2): Multi-processing

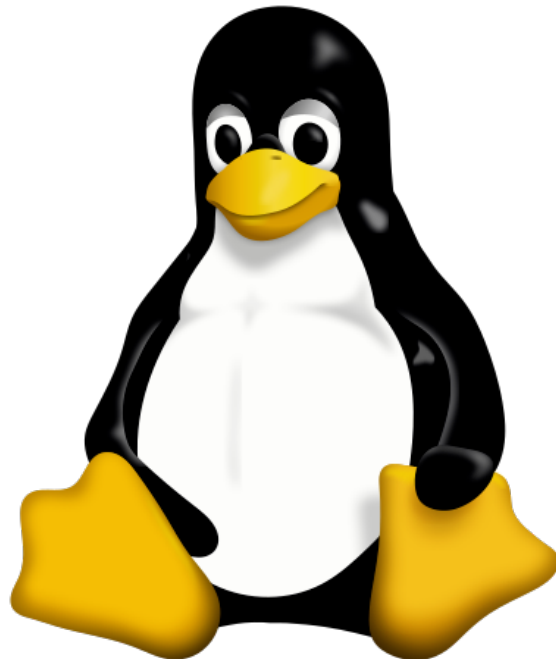
- **Version 2.0** of the kernel includes support for **SMP**, or **symmetric multiprocessing**.
- The kernel could now make proper use of systems with multiple CPUs.





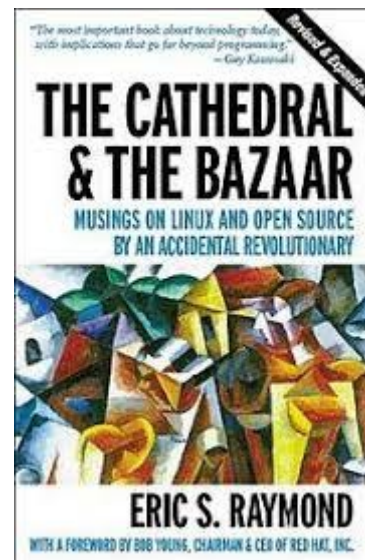
# 1996 (2/2): Tux

- **Tux the Penguin** was chosen as the kernel's mascot.
- His name is short for "**Torvalds' Unix**".



# 1998: Industry acceptance

- **IBM, Compaq, and Oracle** all announce support for Linux for their products.
- Development on **KDE** begins.
- **The Cathedral and the Bazaar** by Eric S. Raymond is published.



# 1999: The DE Wars Begin

- Development begins on **GNOME**.
  - **KDE** still depended on proprietary Qt.



# 2004: Our new overlords

- **Ubuntu** is released, commercially backed by **Canonical**, and based on Debian GNU/Linux.
- Relax **Debian's free software guidelines (DFSG)** a little, and put a whole bunch of money behind it, and suddenly...
- It's hard to get accurate stats, but it's looking like **Ubuntu** is the most popular Linux *ever*.  
...unless you count **Android**.



# 2012: Unquestioned dominance

- Linux-based operating systems now earn more server revenue than all other Unixes *combined*.
- This included both proprietary (AIX) and open source (BSD) flavours.
- To say nothing of its overwhelming clout in **mobile operating systems...**
- Some people are surprised this happened so *late*. Unix ran a lot of big business...it still does.

# Present day: Into the cloud

- **There is no cloud. It's just someone else's computer.**
  - And it runs **Linux**.
- Linux runs **over 90%** of “the cloud” workload.
- Linux **has won, overwhelmingly**, and even The Beast of Redmond must now kowtow, if it wants to stay relevant.

# Year of the Linux desktop?

- **Who cares?**
- Desktops are *so* 2010, bro.

Your personal computing is **already** done overwhelmingly (and increasingly) by Linux computers.

They're just not actually sitting *on* your desk...

# Questions?

- Wikipedia's "History of Linux":  
[https://en.wikipedia.org/wiki/History\\_of\\_Linux](https://en.wikipedia.org/wiki/History_of_Linux)
- Eric Raymond on idealism vs pragmatism:  
[http://www.catb.org/esr/writings/taoup/html/hackers.html#linux\\_reaction](http://www.catb.org/esr/writings/taoup/html/hackers.html#linux_reaction)
- Email: [tom@sanctum.geek.nz](mailto:tom@sanctum.geek.nz)
- Website: <https://sanctum.geek.nz/>