Edit Manage Stats

Ingo Steinke
Posted on Mar 24 • Updated on Apr 19

Another Unnecessary Post in Favor of Linux?

#watercooler #linux #devlive #discuss

In a recent discussion about concepts and technologies that you've changed your mind about I remembered only a few technical tools that I liked from the start and still do, including Linux, HTML, and the GIMP. But I used to be quite skeptical about many new technologies when I first learned about them, including computers and Windows, or graphical point-and-click interfaces in general. Everything used to be so quick and simple just keeping my 10 fingers perfectly positioned above my mechanical keyboard!

What's something you've changed your mind about?

Isaac Lyman • Mar 23
#discuss #career

In the days back then, computers still felt quite special and it was a privilege to have access to or even own a computer. Same with software. As students on a limited

budget, we craved for the free 30-days-evaluation versions of the latest software, hoping that our systems would meet the hardware requirements.

Text Based Roots of my Computing Story

As a kid, I did my first BASIC programming experiments with a language of the same name that felt like some kind of magic natural language parser unlike anything before, maybe the same way that younger developers might feel about chatGPT now. I like the feeling of knowledge and control, like a piano player or a writer with a blank sheet of paper, free to imagine anything.

Actually, my first attempt must have been something like

```
10 READ THE INPUT A$
RUN
?SYNTAX ERROR IN 10
```

before my father — the one who bought our first computer — told me to read the manual (an actual book printed on paper). Still, not bad for a first try. `READ` is a valid command (used to retrieve a row from a database, if I remember correctly), and my intended user interaction would have been a brief and elegant `INPUT A$`.

The following (untested) code should repeat the process all over again.

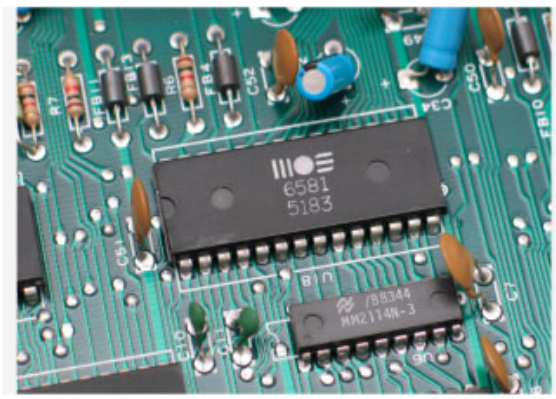
```
10 INPUT A$
20 PRINT "HELLO " + A$
30 GOTO 10
RUN
```

Lacking a photo of my first coding experience, here are some vintage pictures of other boys at that time that might have been me.

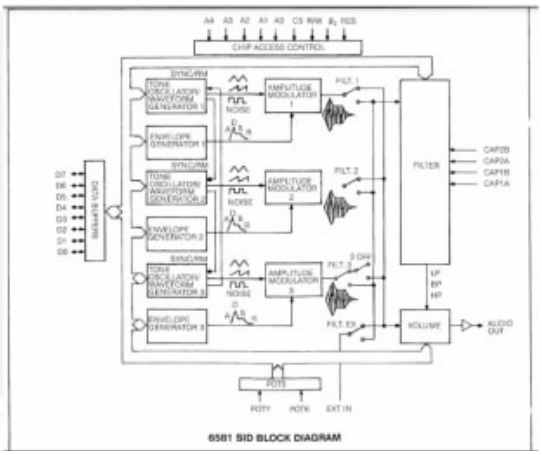
Image sources:

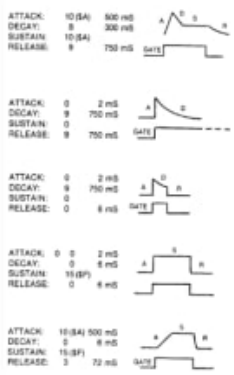
- videospielgeschichten.de/the-computer-kids-die-achtziger-mein-c64-und-ein-computerklub
- www.vintag.es/2019/03/commodore-64.html
- wikipedia.org/wiki/IBM_Personal_Computer

Home computers were able to display graphics. There was a palette of 16 pre-defined colors and a very limited resolution. They could also play music, in a unique synthesizer sound brought to perfection by digital composers like Rob Hubbard. Most kids used home computers as a playstation bought with the excuse of wanting to learn to code.



MOS Technology SID – Wikipedia





Wikipedia: MOS Technologies 6581 SID Sound Chip, circuit and sound synthesizing diagrams.

Graphical user interfaces felt something like a nice gimmick. I was fascinated by the graphics, trying some kind of freehand drawing using all the luminescent colors so unlike crayons on paper.

But using a GUI for coding seemed like a step back, seemingly giving up freedom and efficiency to become a user trying to make sense of the new possibilities. Why would anyone want to see a preview of bold text, font size, or a typeface? Hard to imagine that I would become a web developer, even more so as there was no typography in the early web either.

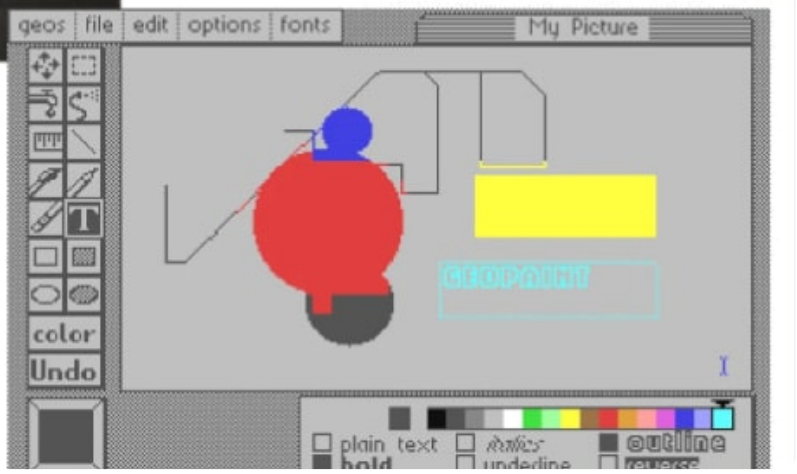
Computers for the Masses

Then computers became more commonplace and the internet evolved, introducing the World Wide Web with its graphical browsers, images, and clickable hyperlinks. Universities and internet cafés provided the opportunity to use different machines to "surf the web", and I found out about UNIX one day when the only free workstation was the "server" machine running IBM AIX. Computers by Apple, a pioneer of so many important technology, used to be a very special niche device for graphic designers, much like a SUN or Silicon Graphics workstation.

File:Commodore Founder Jack Tramiel and VIC20 Product Manager Michael Tomczyk celebrated the first million units i...

Get this image on: Wikimedia Commons | License details

File:SGI Indy.jpg - Wikimedia Commons



geos desktop for c64

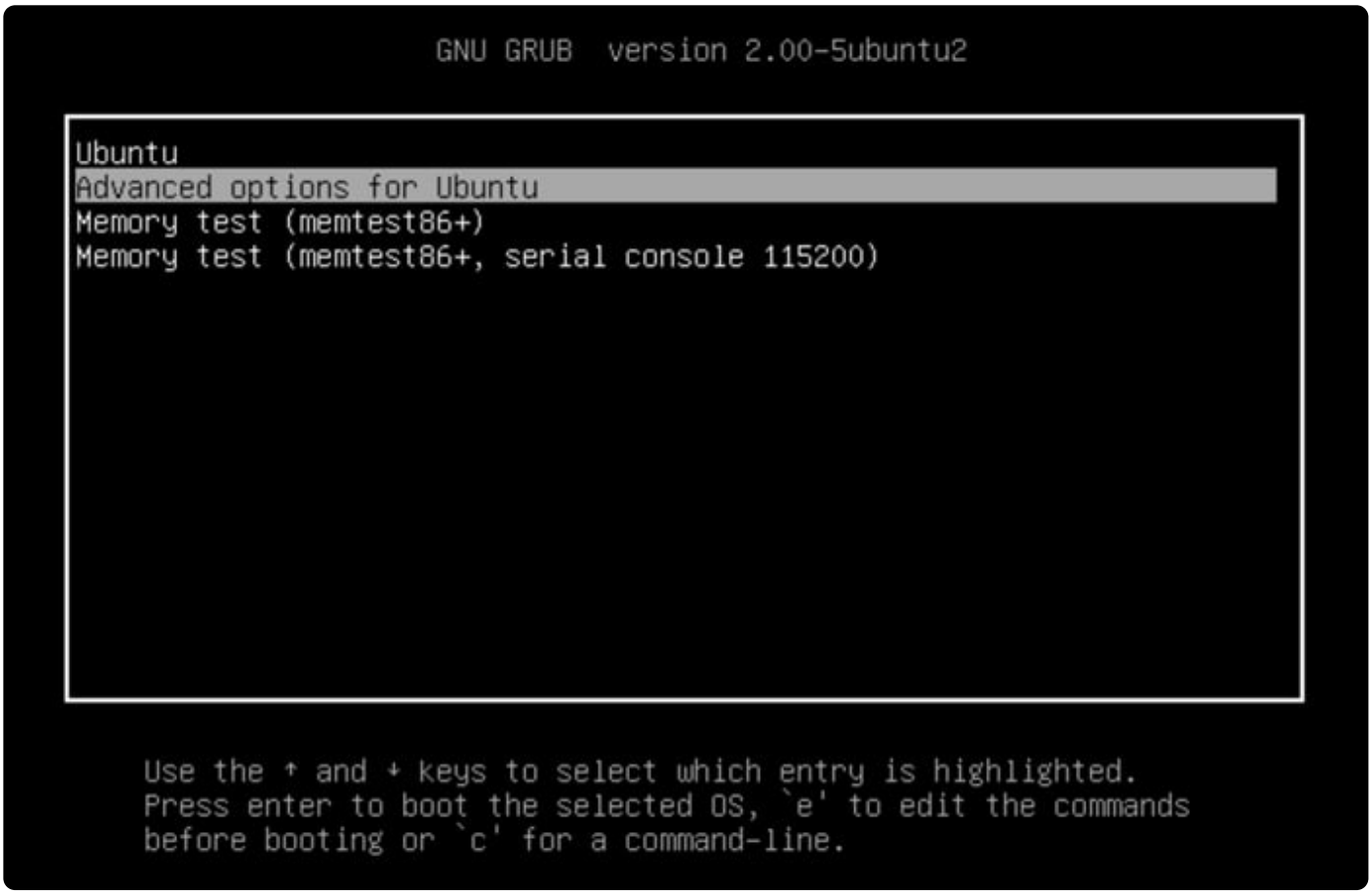
I started to get bored and frustrated about the limitations of my Windows PC at home. Personal computers, with their modular hardware architecture and affordable components from "compatible" producers and second-hand repair stores, proved to be the next computers for the masses, and the Windows / MS-DOS operating system was hacker-friendly enough for experimenting. But, possibly due to business requirements, Windows became more and more "professional" and thus less configurable with each update, while still unprofessionally degrading over time, notorious for its errors and crashes destroying data or the whole operating system.

An Alternative Open-Source Operating System

One day my father told me about a new alternative operating system that he had read about, an open-source version of UNIX, free to use, maintained by a non-profit community.

The early Linux distributions were still far from a Windows replacement and originally never intended to be anyway.

Luckily, I had not forgotten about my text-based "hacker" roots before graphical user interfaces, so it did not stop my enthusiasm for Linux that I did not manage to start its graphical user interface on my PC.



I kept using both Windows and Linux in parallel, which required a reboot to change the operation system. Ever so often, I would use Windows for every day work and study, but also for graphic image editing and multimedia experiments. Photoshop, Bryce, Dancer DNA, Rebirth, and other more or less obscure software inspired me and my friends for a lot of creative work and leisure. I rarely played computer games anymore, as I preferred a multimedia jam session to shooting virtual characters in a clunky 3D landscape.

PC vs. MacBook — a Professional Choice?

Fast forward into professional life, companies often ask new team members if they prefer to use a PC or a Macbook. Most of them still have no default Linux option, but of course there will be colleagues opting for a Windows PC and install Linux unless it's officially forbidden.

Operating systems have changed and conformed, integrating popular trends over time, so it does not matter that much anymore. I can do my work on a Mac, on a Windows PC, or on my Linux laptop. But there are still some fundamental differences and so it still matters at least to me.

Here are some portable devices by different manufacturers. There is an old, but functional, MacBook that dates back at least to 2013, a refurbished Lenovo ThinkPad that boots with Linux Mint or Microsoft Windows, a Linux laptop produced by TUXEDO, a budget smartphone running Android, and an outdated Apple iPhone that I sometimes use for testing my websites on real-world devices.

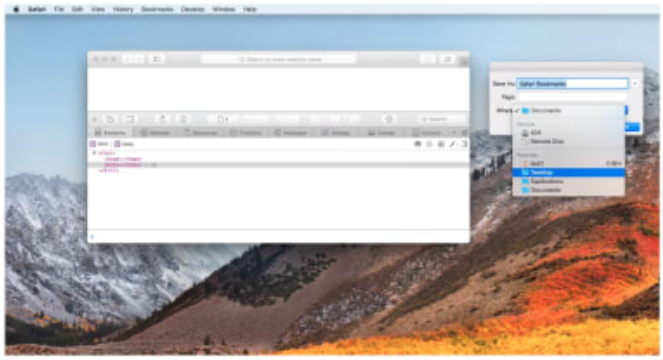
Advantages and Disadvantages of Mac vs. Windows vs. Linux

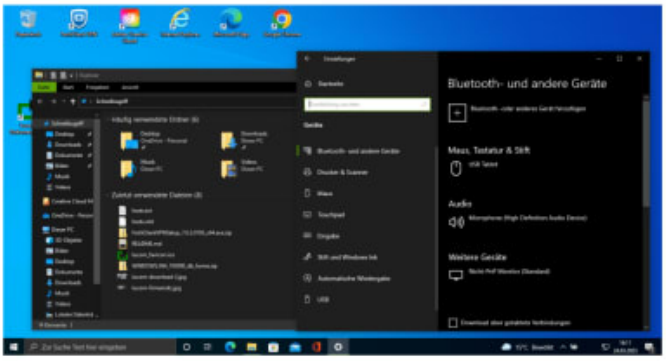
Both Apple / Mac(book) and Windows machines are known to be popular commercial products with professional support and high hardware quality. While Linux is neither that popular nor known for it, you can get all of that with a professional Linux machine, preinstalled with a mainstream distribution, commercial drivers and professional support. Companies like Red Hat and SUSE have established business models based on and around open-source software proving that there are alternatives to closed-source products and marketing-driven companies.

Apple has successfully managed to transform their operating systems, or at least the desktop version, into a stable software based on a UNIX system with a very consistent and elegant user interface. On a software level, MacOS and Linux have converged in a way that we can share common code with Docker and shell scripts in a team of Mac and Linux users without worrying too much about compatibility.

Windows Incompatibilities

Now Microsoft Windows has become the unlikely development environment. Despite their efforts with WSL and open source software like the brilliant Visual Studio Code (which also runs on Linux and Mac!), don't trust a Docker setup to work on a Windows machine, at least not at the same speed. Although I know a few people who love their Windows setup and were probably either lucky or invested a lot in top hardware and a perfect configuration, in the past years it has always been Windows where something did not work as expected. Same with the infamous Internet Explorer, but to be fair, they managed to replace it by the new Edge, and Internet Explorer used to be a motor of innovation before Firefox, Chrome, and Safari had even been around.





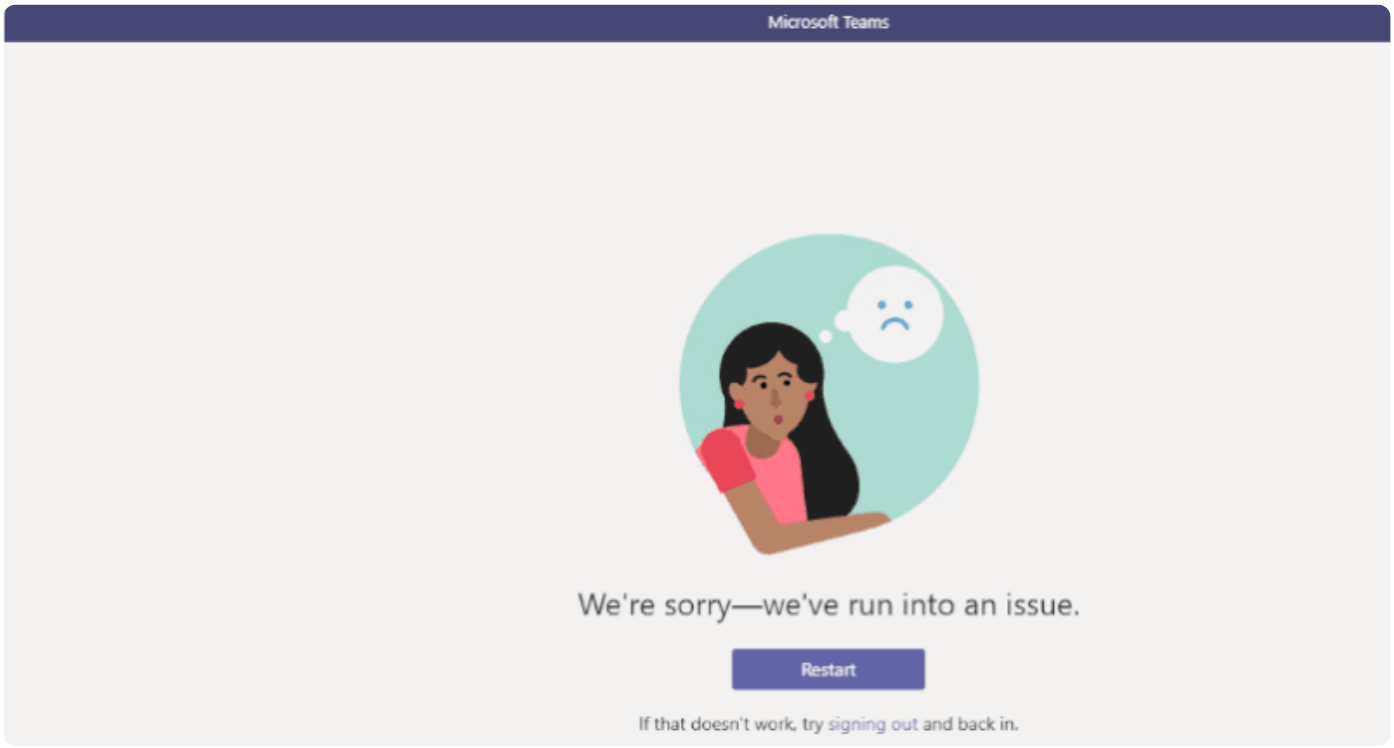
Steve Jobs might have been a difficult character to say the least, but Apple got some things right that Microsoft and many other companies didn't. Microsoft invented the "pocket PC" long before Apple's iPhone, but it was the latter that fundamentally changed the way we use the internet and do our work on the go without having to sit on a chair in an office from 9 to 5.

I chose a Macbook at my last two jobs as an employee and I was quite happy about it. As I said, the hardware is great, the user interface looks nice, and Macbooks are quite lightweight compared to a Windows PC with a similar performance.

Vendor Lock-In as the Greatest Disadvantage

Maybe the greatest disadvantage of Apple is its vendor lock-in. As Apple users, we depend on Apple's decisions. The current MacOS version still offers a lot of choice, or rather ways to work around the recommended tools and limitations. But why do we have to install XCode and agree to its license if all we want is use regular GNU command line tools, git, and homebrew? On a Macbook, we aren't forced to use Safari as the only browser, but as iPhone users we are. And while the old Macbooks, thanks to their supreme hardware and high quality, are still doing their job after ten years, we can't upgrade to the latest operating system anymore, missing out on security updates and the latest Safari browser version.

But in my opinion, Windows is even worse in every aspect. The UI has changed a lot, replacing the old look and feel with a clunky design with some very impractical UX degradations like opening the system settings or a file browser in a very small window despite the large icons and spacing so that users need to scroll to find something that could have been visible at first sight at least on a large monitor. The latest Windows versions have removed a lot of the configuration possibilities that used to be available in earlier versions, so we have to love or leave the way they intend their product to look and feel on my computer.



Microsoft has been praised for its accessibility and internationalization achievements, but as a native speaker, I always found there German documentation, captions, and error messages misleading and hard to understand. Later I found out that also applies to the original English messages. I always cringe when someone sends me a Teams meeting invitation. Receiving it in any other email client apart from Outlook, it is sometimes impossible to guess the time and date of the meeting but if that shows up and you even manage to accept and save the date in your calendar, there is still the actual meeting. How to be sure which Teams client to use and which account to log into? Maybe this works on single-user Windows client with a corporate email address within the same company or organization. But who knows? I am not going to switch back to Windows to fix your usability bugs. It's only a matter of time until Teams will follow Internet Explorer into the hall of shame of software that nobody will miss (together with Flash and a bunch of other products bought and destroyed by Adobe, their kindred spirit).

Linux as a Choice for "Non-Technical" Users

Let's be honest, Linux has its issues, and without professional support, I would not recommend it for everyone. But then again, it does not matter that much on a basic level anyway. If all you need is a text editor, a web browser, and an option to print a document every now and then, Linux might still be the best choice even for "non-technical" users. Go for a mainstream distribution like Mint or (L)Ubuntu, put Firefox and Libre Office shortcuts in the main menu, and you are set!

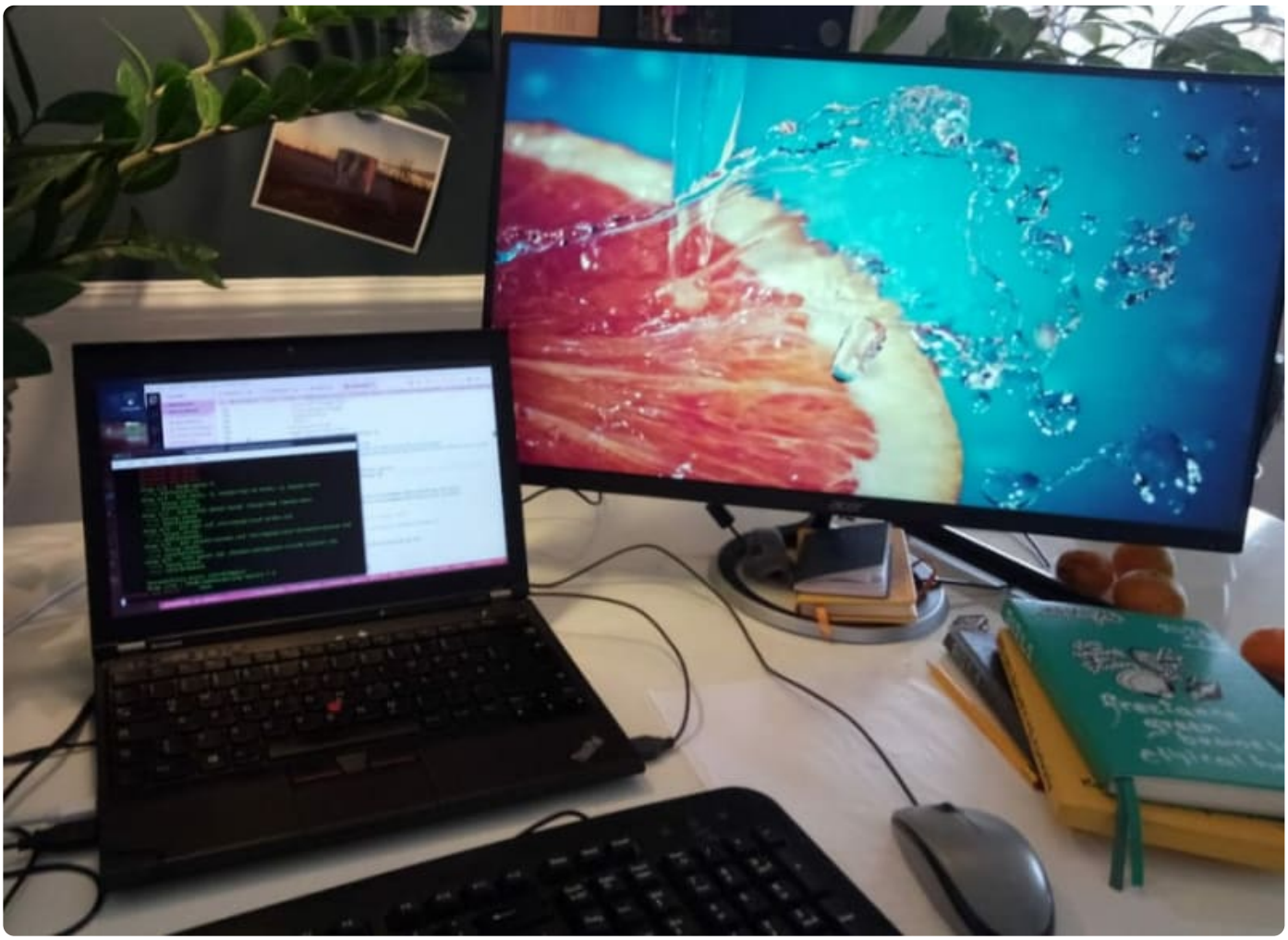
There is still a lot of (mostly commercial) software that does not run on Linux, at least not officially. We can try and use WINE which enables quite a lot of Windows software to run on a Linux system, and we can use virtualization like VirtualBox or VMware to provide a 99% perfect emulation of an actual Windows computer, but that's such a waste of performance that some software like Adobe Photoshop is still not guaranteed to work properly. Hardware intensive multimedia tools for musicians, film makers, and gamers should be used in the environment that it's intended for, and the same goes for developing native Windows software of course.

But on a more modest and average level, we can do games, graphics, music, and even video editing on a modern Linux computer. Many distributions already come with some default choices that do their job quite well.

Linux Support for Old Hardware

Maybe the greatest advantage of Linux is the support for old hardware. If you want to save money and the environment by using your electronic devices for a very long time, Linux is for you! You can take a very old PC, either desktop or laptop, install Linux, and get the latest versions of the most important software, and of course every necessary security update of any relevant system component. Well, mostly. Some software stopped supporting 32-bit architectures, but as most Linux software is open-source, we could make an effort and compile a binary that runs on our system.

Reduce, Reuse, Repair and Donate!



To show and prove, I have reused some older machines thanks to Linux, either to donate them for charity or for my own personal use. I even developed my own website on an old refurbished ThinkPad before I got my current top notch TUXEDO InfinityBook Linux laptop. You can read more about reusing laptops in my story about low-waste tech: reduce, reuse, repair and donate!.

Low-Waste Tech: Reduce, Reuse, Repair and Donate!

Ingo Steinke • Mar 12 '21

#sustainability #repair #linux #lineageos

Conclusion: maybe this is yet another unnecessary post in favor of Linux to add some continuity and consistency to my DEV blog, but I hope that it might inspire you or that I might even be charging an open door. So have fun with Linux — or leave a comment to tell why why you disagree! 😉

Sustainable Low-Waste Tech (3 Part Series)

1 Low-Waste Tech: Reduce, Reuse, Repair and Donate!

2 **Another Unnecessary Post in Favor of Linux?**

3 Sharing a Printer in a WiFi Network

Top comments (4)

Ben Sinclair • Mar 25 • Edited

> a file browser in a very small window despite the large icons and spacing so that users need to scroll

This is worse on MacOS, because there the scrollbars are hidden by default, and unless you *manually* tell it to sort your icons, there's a pretty good chance the thing you want to find is way, WAY off the screen to the right somewhere, so you don't even realise the icon exists.

geraldew • Mar 25

A nice read, thanks for writing this. Admittedly I'm a similar (earlier?) vintage, so I fully "get" that idea of having a computing mentality based in the text world rather than full colour pixelated GUIs.

Reading through it, there are plenty of spots where I want go "that's not quite right" but of course, this is one of the main facets of (what I still call) the personal computer revolution - that freedom to have quite different personal experiences.

Having recently returned to writing an application outside of my employment, it has been interesting to feel my reasons for giving it a GUI rather than command-line interface. With the main one harking back to my very early experiences of command line usage and programming - which is: that if you don't already have a manual for what the commands are, then you can hardly do anything at all.

Nonetheless, my past has experiences of indeed learning things via the error messages. I can recall having a CP/M version of Prolog for which I had no manual and very little knowledge of the language.

All these years later, the situation is not much different when I'm asked to write T-SQL (the Microsoft dialect) and having to net search for each syntax quirk despite being able to write reams of Teradata SQL as air code before executing any.

Also, in getting a bit old, I forget even my own abbreviations for clever features that I've given my hand-crafted tools, meaning that I also want them to have a GUI that shows me what all the options are, replete with hover-over descriptions and links to documentation.

Long before it became my profession, I loathed the *hidden secrets* aspects of computing - it seemed to operate the same way that classism had hidden options from me during my formative years. That's obviously a wider topic.

So while I have a nostalgia for the learning that comes tackling the mysteries of computing - acquiring knowledge from puzzle solving - I don't wish it upon others.

That all said, it seems that the tools I make (sadly, in an environment from where I cannot share them here) no matter how apparent I try to make them, remain elusive for most people I show or demonstrate them to. Perhaps that's irony.

Jeansen • Mar 29

I started with Windows, been using Linux now for almost a decade. At work I have to use a MacBook. Decisions for the latter one are fine with me because MacOS simply is (not yet) a big target for all sorts of malicious attacks. Windows, I do not use anymore. There was a point where I felt blocked doing regular development. And here comes Linux. What I like about it is, that on one hand there is no registration, no fee, no activation code .... you simply install and run it. On the other hand, the philosophy makes it quite easy to backup, restore, update, upgrade or transition a system. Yes, YOU HAVE TO UNDERSTAND LINUX. It is not a click-and-fund system. So, with Linux, one needs patience. It's like learning a language. You cannot expect to speak any language fluently after one week. My Debian system is almost a decade old - from the time, when I started.

With it, I learned and grew. Show me a Microsoft or Apple system that you upgraded throughout a decade without breaking things or having to reinstall it, taking it from one computer to the next, and the next.

Ingo Steinke • Mar 29

I never took any Linux installation to another machine. Quite contrary, I try to become as independent from specific machines as possible. Maybe that's due to using Windows for so many years and switching companies and projects.
I used to have a backup of my "best" Windows (95) setup, including some drivers, tools, and settings that were hard to find or remember. Now I have a public git repository to persist certain preferences, and sometimes when I google a specific problem, I remember that I used to know the solution and even wrote a blog post about it.

As I merely touched my Linux history, I tried quite a lot of different Linux distributions over the years, starting with Slackware, SUSE, and several less known alternatives, until I found Debian, Ubuntu, and Mint as end-user-friendly desktop distributions.

Yes, you still need some patience as a developer, but so you do on a Mac or on Windows! Docker, git, shell scripts, and all the DevOps details are not easy to learn anyway. But I think that Linux has become very usable for ordinary people. And Windows gets in the way of those regular users and office people as well. It's just that people take it for granted and get trained "how to use a computer" which mostly means "how to make sense of the latest Microsoft software".