



Understanding the energy use of Firefox

With less power comes more sustainability

Florian Quèze

February 4, 2023 - FOSDEM

Table of contents

- **Why?**
- **Understanding power use**
 - locally
 - in the wild
- **Improvements**



**Why do
we care?**

Why? User experience!

- Noisy fans
- Hot laptops
- Battery life



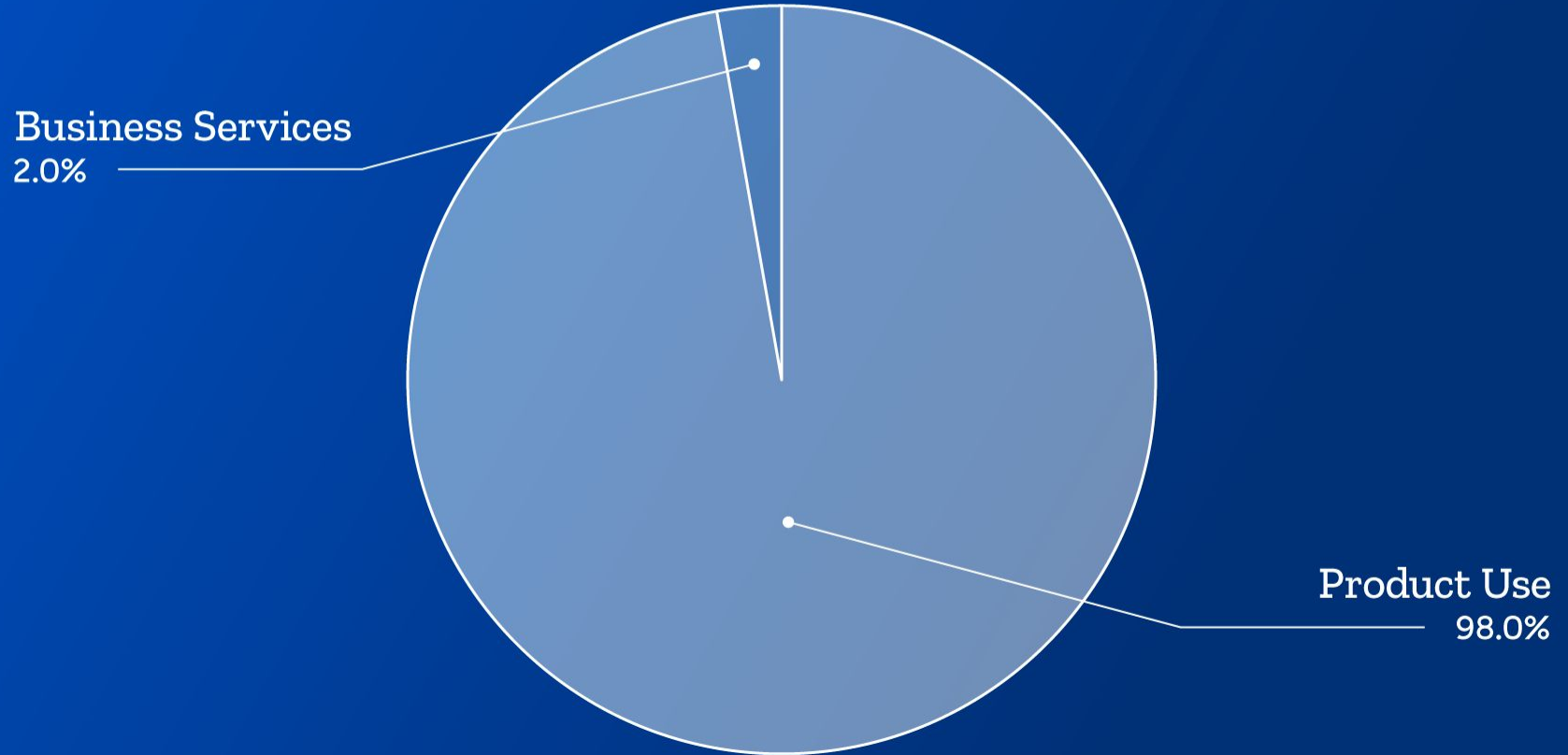
Why? Sustainability!

Mozilla made [climate commitments](#):

- being carbon-neutral.
- reducing its [GHG footprint](#) year over year
- **leading openly by sharing materials, tools, and methodologies.**
- exploring approaches to develop, design, and **improve products from a sustainability perspective**



Emissions Distribution 2019





Understanding local power use

How Firefox uses power

- CPU time
- GPU time
- CPU core wake-ups
- Network packets



How Firefox wastes power

- Using too much CPU time
- Waking up threads too often
- Invisible animations
- Background activity



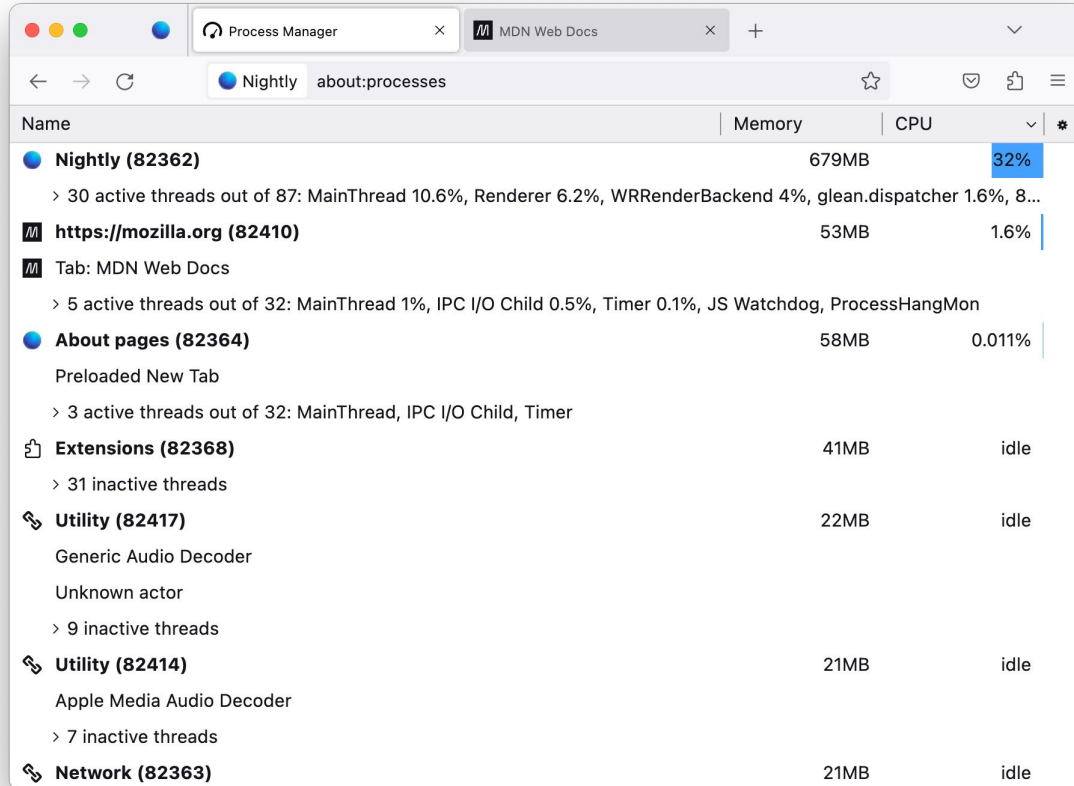
Understanding local power use

Troubleshooting excessive power use:

- One or more cores 100% used
- idle Firefox processes that are not really idle.



Firefox task manager



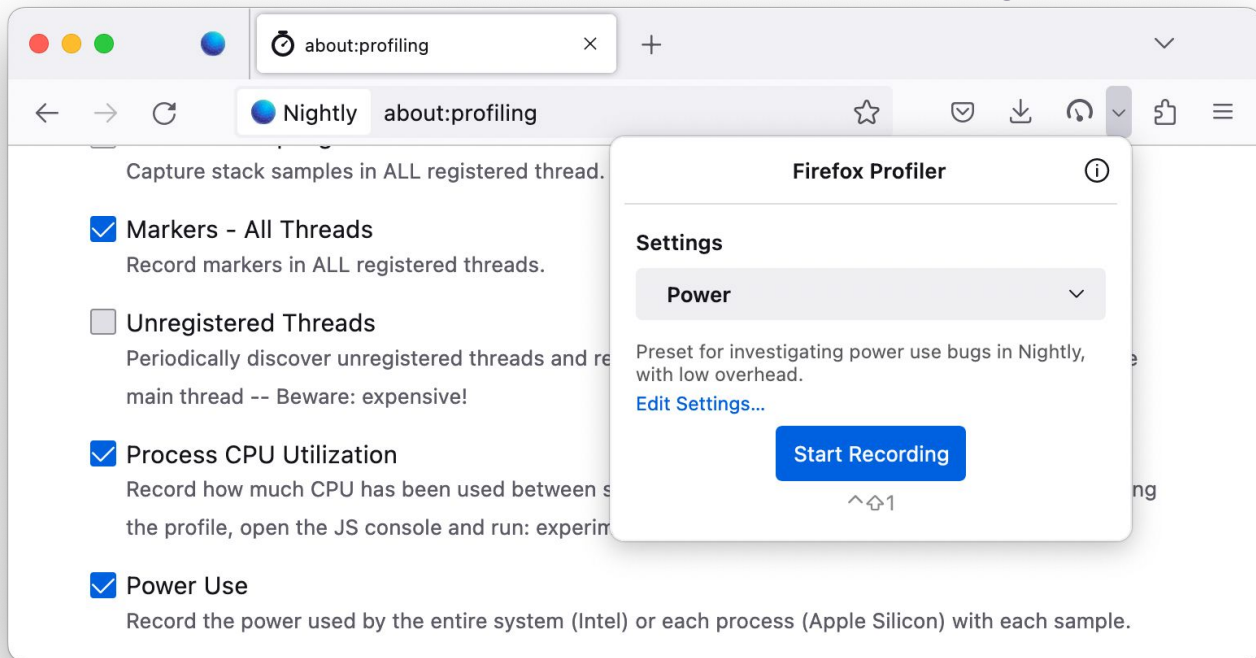
The screenshot shows the Firefox Task Manager interface. The browser window has two tabs: 'Process Manager' and 'MDN Web Docs'. The address bar shows 'Nightly' and 'about:processes'. The main content area displays a table of processes with columns for Name, Memory, and CPU. The 'Nightly (82362)' process is highlighted with a blue background and shows 32% CPU usage. Other processes include 'https://mozilla.org (82410)', 'About pages (82364)', 'Extensions (82368)', 'Utility (82417)', 'Utility (82414)', and 'Network (82363)'. Each process entry includes a list of active threads and their respective CPU usage percentages.

| Name | Memory | CPU | * |
|---|--------|--------|---|
| Nightly (82362) 30 active threads out of 87: MainThread 10.6%, Renderer 6.2%, WRRenderBackend 4%, glean.dispatcher 1.6%, 8... | 679MB | 32% | |
| https://mozilla.org (82410) Tab: MDN Web Docs 5 active threads out of 32: MainThread 1%, IPC I/O Child 0.5%, Timer 0.1%, JS Watchdog, ProcessHangMon | 53MB | 1.6% | |
| About pages (82364) Preloaded New Tab 3 active threads out of 32: MainThread, IPC I/O Child, Timer | 58MB | 0.011% | |
| Extensions (82368) 31 inactive threads | 41MB | idle | |
| Utility (82417) Generic Audio Decoder Unknown actor 9 inactive threads | 22MB | idle | |
| Utility (82414) Apple Media Audio Decoder 7 inactive threads | 21MB | idle | |
| Network (82363) | 21MB | idle | |



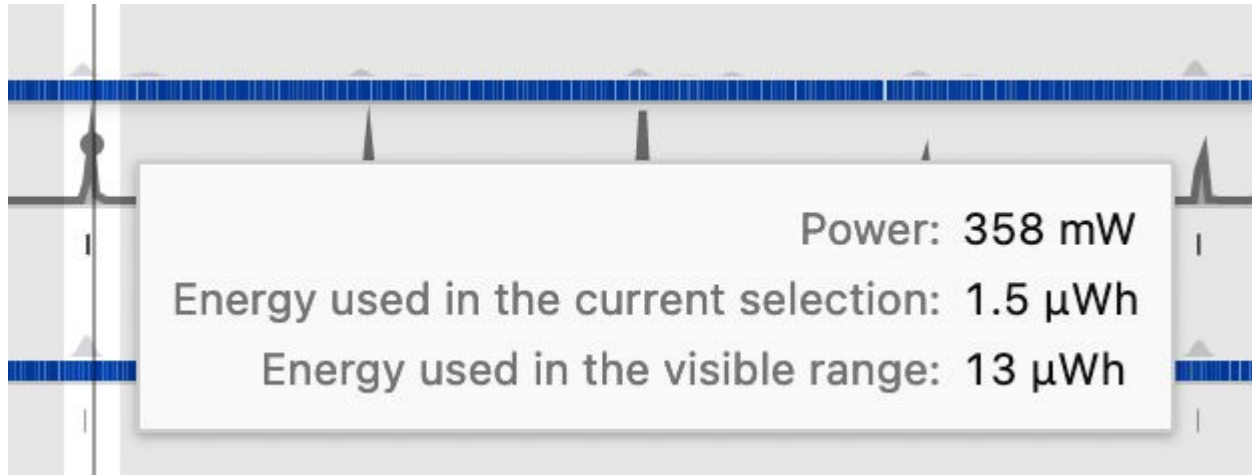
Firefox profiler

The Firefox Profiler now supports power profiling.



Firefox profiler - power profiling

Measure tiny things:



Ever wondered how much power it takes to blink the caret in the address bar?

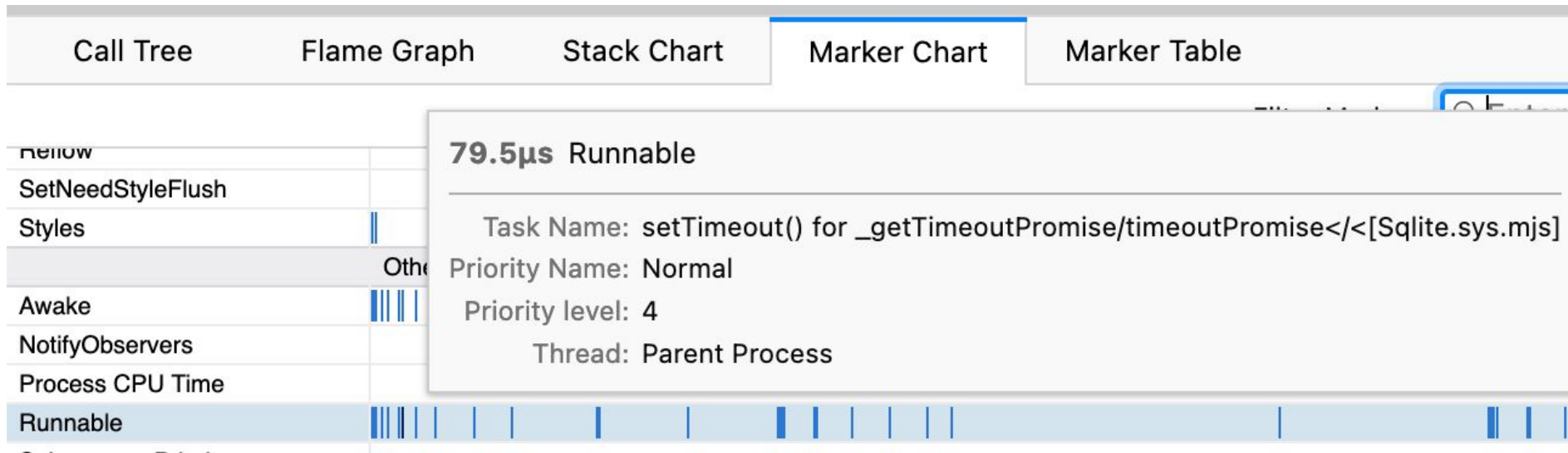
Now you can know!

<https://share.firefox.dev/3U8hLgp>



Firefox profiler - markers



“Awake” and “Runnable” markers show thread wake-ups



Firefox task manager

One click profiling

The screenshot shows the Firefox Task Manager interface. At the top, there are window control buttons and a title bar with 'Process Manager' and a close button. Below the title bar, there are navigation icons and a search bar containing 'Nightly' and the URL 'about:processes'. The main content area lists several processes:

- Nightly (82362)**
 - > 16 active threads out of 86: MainThread 0.7%, (41382907) 0.4%, (4
- https://mozilla.org (82410)** (selected) 
- Tab: MDN Web Docs** 
> 31 inactive threads
- About pages (82364)**

A tooltip is visible over the 'Profile icon' of the 'https://mozilla.org (82410)' process, containing the text: 'Profile all threads of this process for 5 seconds'.





Understanding global power use

New telemetry probes

We added data collection for the following:

- Total CPU time used
- Total GPU time used
- Total number of thread wake-ups
- Breakdown per process type
- Breakdown per thread name (Nightly channel only)



Estimating our footprint

- Firefox's daily global resource use:
 - 60-80M hours of CPU
 - 15M hours of GPU

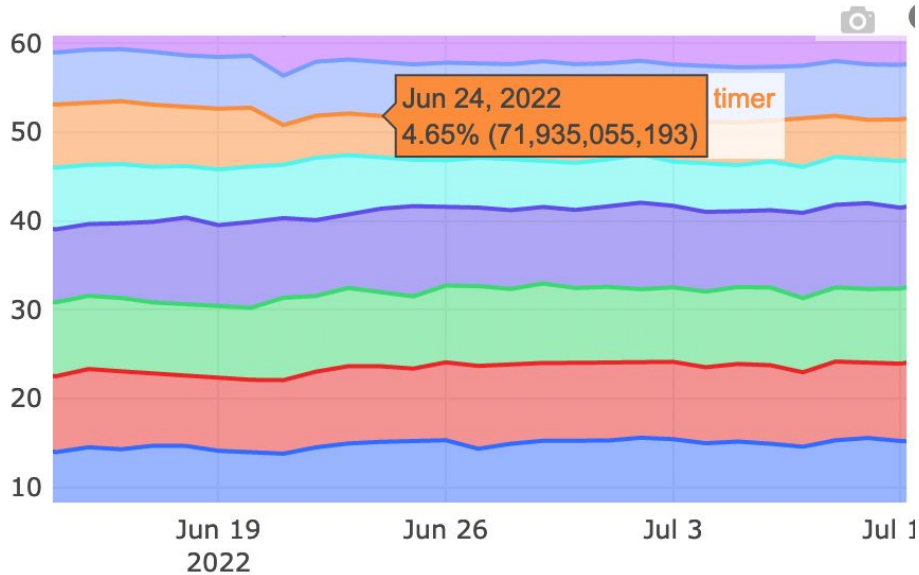
Can be converted to CO₂e using CPU specifications and electricity carbon intensity by country.

Roughly equivalent to the electricity production of a small thermal power station, or 50,000 solar roofs.



Verifying fixes

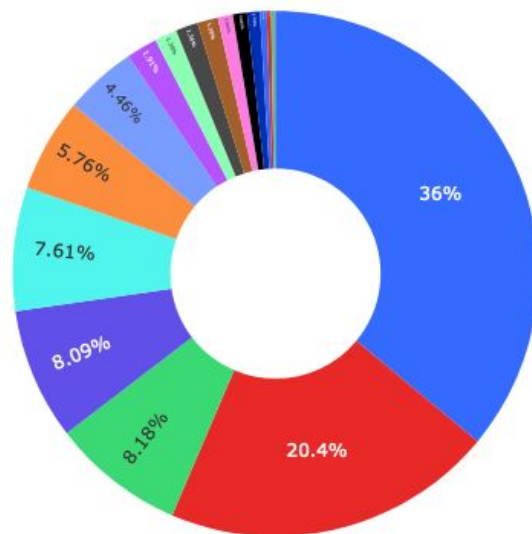
Percentage over time – Thread wakeups per thread name



On June 20, [bug 1767396](#) made the timer thread drop from being 7% of our thread wake-ups to 5%.



Testing assumptions



Understand the potential impact of changes:

Eg. background content processes use about 8% of our CPU time





Improvements

Bug fixing

We fixed 26 power bugs since Firefox 95



- Removed repeating timers
- Stopped hidden or bogus animations
- Avoided pointless thread wake-ups
- Fixed edge cases with excessive CPU use.

Thanks to everybody who helped!



Using efficient cores on Windows 11

Used power profiling to measure the impact of changes:

| | | | | | |
|--|-----------------|---|--|---------|---|
|  | Firefox Nightly | Efficiency mode  | 0% | 21.7 MB | 0 |
| | | | This process is in efficiency mode to limit resources used by the process. | | |

The power used by a content process using 100% of a CPU core drops from 10W to 2W when the process is put in “efficiency mode” (EcoQoS):

<https://share.firefox.dev/3FJoAkc> ([bug 1796525](#))



Preventing regressions

[Bug 1742842 - Ensure VSync is disabled at the end of automated browser chrome tests](#)

“We found multiple times by accident steps to reproduce bugs that cause vsync to remain enabled forever (or until a browser window is closed). We should leverage our large existing test suite to detect cases like this faster.”



Next / ongoing

Areas where we can improve efficiency even more:

- Background tabs
- (grouping) timers
- Invisible media and animation
- Fully occluded windows
- Long user idle time



Ideas to experiment with

- reducing frame rate

```
gfx.display.max-frame-rate = 30 (Hz)
```

- disabling video autoplay

```
media.autoplay.default = 5 (Block Audio and Video)
```



Thanks! Questions?

- Share ideas, #power-usage:mozilla.org on Matrix.
- Questions: florian@mozilla.com

