



CPU frequency scaling for gaming workloads

Status of the amd-pstate epp_boost RFC and next steps upstream



LINUX
PLUMBERS
CONFERENCE

PRAGUE, CZECHIA / OCT. 5-7, 2026

David Vernet (Anthropic) <void@manifold.com>
Gaming on Linux MC

Goals for this session

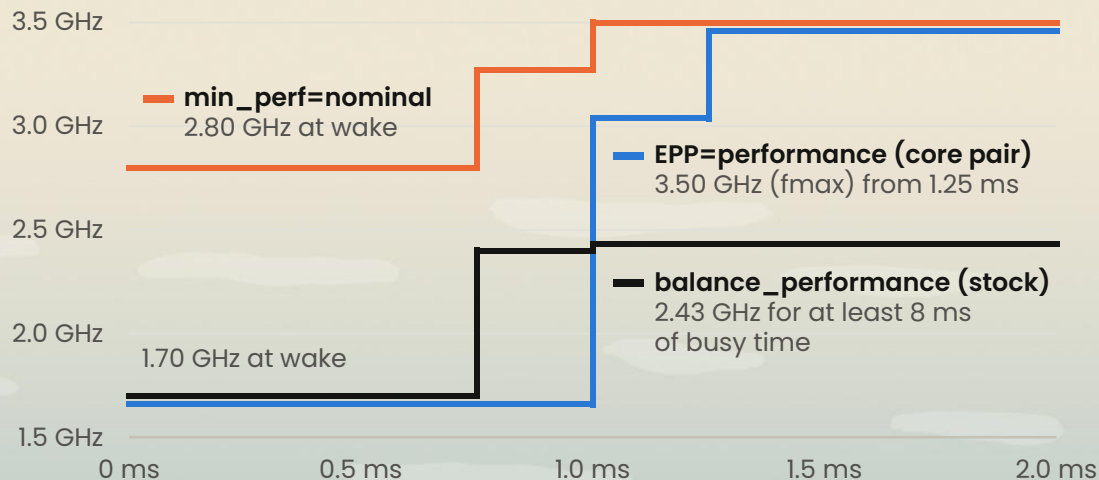
1. Discuss the best approach to power management on devices with tight power budgets, complex topologies and several sources of power draw.
2. Should we plug into Wine to hint which threads matter?



With stock EPP, short sleeps keep a core below fmax

- In Civ VI on a Steam Deck, a busy core sits at 2.43 GHz. fmax is 3.5 GHz.
- Short sleeps reset the firmware's demand estimate.
- The cost shows up in frame-time tails and load time, not in average fps.

Median effective clock by time since wake on Van Gogh



AMD maintainers asked for a different architecture

What the RFC does

- It samples core busy time every 10 ms.
- A core that is 50% busy gets EPP=performance.
- Civ VI 1%-low fps rose 29% to 38%.
- It costs about 1 W while cores are busy.

What the review asked for

- No module parameter.
- No overriding of a user-set EPP.
- A governor-level mechanism, not a driver heuristic.
- The requested data is not yet posted.

[\[RFC PATCH 0/4\] cpufreq/amd-pstate: Per-core EPP boost for recently-busy CPUs](#)



Guided mode has the best tail but at an always-on cost



- Guided mode sets a min_perf floor and holds EPP at performance on every CPU.
- At light load it draws 0.6 to 0.9 W more than active mode. The RFC draws at most 0.2 W more.



1. Why not just use guided mode?

Guided mode asks for a frequency

- schedutil sets a min_perf floor.
- The scheduler cannot see GPU power.
- EPP is fixed at performance.

EPP asks for a QoS (quality of service)

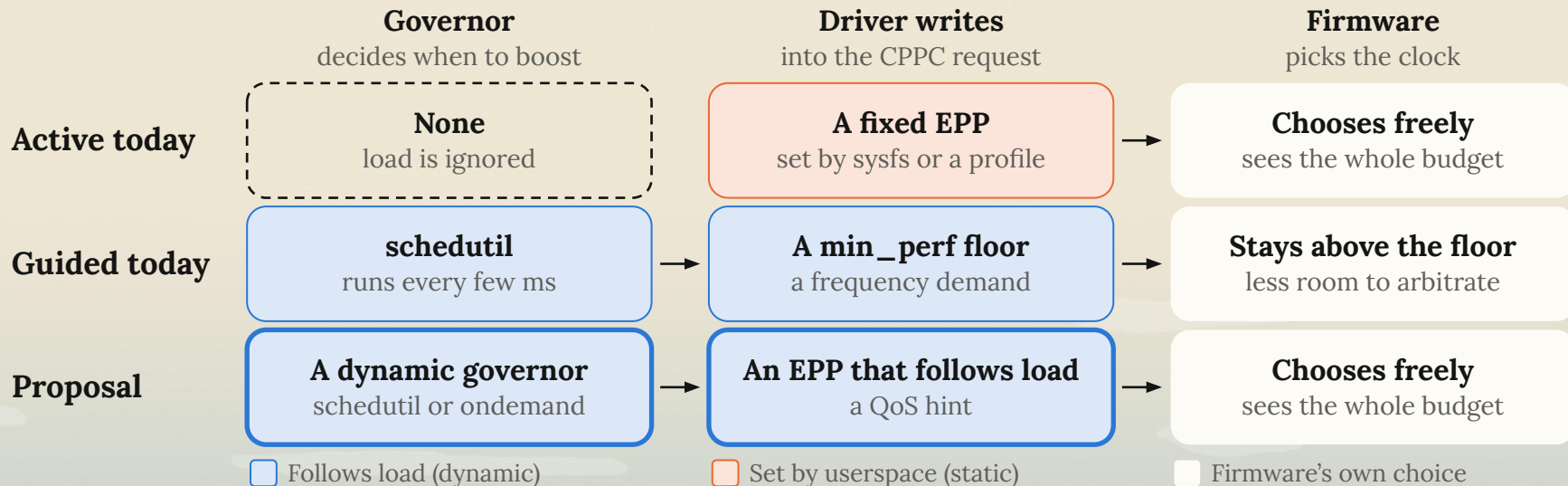
- It is a preference, not a frequency.
- Firmware sees the whole power budget.

On a handheld

The CPU and GPU share one power budget, and only firmware sees both. Guided mode overrides it at all times, even at light load.



Proposal: let a dynamic governor drive EPP

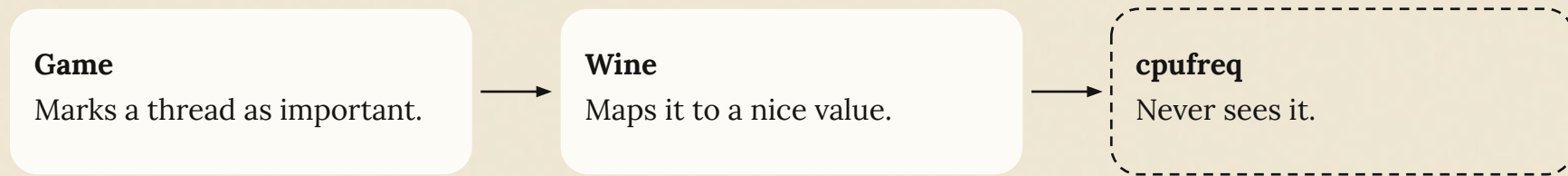


- Combines guided mode's governor and active mode's hint
- Policy stays in cpufreq, and power arbitration stays in firmware.

Prateek suggested dynamic-governor support for active mode in [his review of the RFC](#).



2. Aside: should we be leveraging thread hints?



What Wine could do

- It could set UCLAMP_MIN instead.
- A prototype cut work time 41% on a 7950X.
- The proposal carries that clamp to EPP.

OpenXR already does this on VR headsets

- An app labels its threads as time critical or background.
- An app requests a performance level for the CPU or GPU.



Reference material

Appendix



LINUX
PLUMBERS
CONFERENCE

PRAGUE, CZECHIA
OCT. 5-7, 2026

A busy core on the Steam Deck runs well below f_{max}

- The median clock of a busy core is 2.43 GHz.
- 58% of samples are below 90% of f_{max} .
- About 63 short futex sleeps per second reset the demand estimate.
- EPP=performance in sysfs raised 1%-low fps by 16%.

The data is from a Steam Deck OLED, with interleaved A/B runs and a Welch t-test.



LINUX
PLUMBERS CONFERENCE

PRAGUE, CZECHIA / OCT. 5-7, 2026



The RFC sets EPP to performance on cores that were recently busy

Mechanism

- An `update_util` hook samples C0 residency every 10 ms.
- At 50% busy it sets EPP to performance, with a 300 ms decay.
- Futex waits never set `iowait`, so `hwp_boost`'s trigger cannot fire.

Results on the Steam Deck

- Civ VI 1%-low fps rose 29% to 38% ($p \leq 0.014$).
- Package power rose 1.0 W while busy.
- A `min_perf` version regressed p999 and was dropped.

[RFC PATCH 0/4] `cpufreq/amd-pstate`: Per-core EPP boost for recently-busy CPUs. It was posted to `linux-pm` on July 28.



epp_boost results on the Steam Deck

Workload	Shape	Metric	Off to on	p
Civ VI benchmark	Single-thread CPU-bound	1%-low fps	+29% to +38%	≤ 0.014
vkmark submission loop	CPU-bound with short waits	fps	+6.1%	< 0.0001
Factorio benchmark	Continuously busy	Stall blocks > 1.1 s	4/25 to 0/25	n/a
vkmark (GPU-bound)	94% gpu_busy	fps	+0.1% (no shared-TDP cost)	ns
Power	Civ VI	Package power	+1.0 W (+11-13%); fps/W -10%	≤ 0.011

The data is from a Steam Deck OLED, with interleaved A/B runs and a Welch t-test.



A short sleep resets the Van Gogh operating point

Time since wake	balance_performance	EPP=performance (core pair)	epp_boost (RFC)	min_perf=nominal
0 – 750 μ s	1.70 GHz	1.70 GHz	1.70 GHz	2.80 GHz
750 μ s – 1 ms	2.40 GHz	1.70 GHz	1.78 GHz	3.27 GHz
1 – 1.25 ms	2.43 GHz	3.08 GHz	3.13 GHz	3.50 GHz
1.25 ms onward	2.43 GHz (persists)	3.50 GHz	3.50 GHz	3.50 GHz

- A sleep of a few hundred microseconds resets the operating point, at any C-state depth.
- The 2.43 GHz state then persists for at least 8 ms of busy time.
- EPP selects the ramp target. Only min_perf raises the clock at wake.

Values are median effective clocks over 3×40 s runs with an 8.3 ms idle gap.



v2 must write the EPP hint on both SMT siblings

EPP=performance set on	Busy clock	High-state share	Completion p99
None (balance_performance)	2.37 GHz	0–6%	10.1 ms
Busy thread only	2.37 / 3.27 GHz (bistable)	0 / 53 / 77%	10.1 ms (unchanged)
Idle sibling only	2.37 GHz	8–33%	10.1 ms (no effect)
Both SMT siblings	3.26 GHz (deterministic)	100% (0 low periods in 14k)	7.5 ms
All 8 CPUs	3.27 GHz	100%	7.5 ms

- Van Gogh aggregates the EPP hint per physical core.
- The RFC as posted writes one thread's MSR, which is the bistable second row.
- On a 7950X the busy thread's own hint is enough.

The test is a single pinned thread with an idle SMT sibling. Only the EPP byte differs between rows.



Removing droop does not by itself improve p999

Configuration	Droop episodes	Worst 1%	p999	Notes
active + balance_performance (stock)	$7 \times \sim 4$ s	70–73 ms	75–77 ms	Baseline
active + epp_boost (RFC)	$8 \times \sim 0.4$ s	63 ms	76 ms	+1 W while busy
active + dynamic min_perf floor	0	59 ms	86 ms	Worst p999 (80 / 90 / 86 ms)
guided + schedutil	0	47 ms	51 ms	min_perf follows schedutil (+0.9 W)
passive + schedutil	2	69 ms	79 ms	Mean 7% worse. PELT under-requests.
passive + UCLAMP_MIN on render thread	0	67 ms	76 ms	Requires identifying the thread

- The dynamic floor removes droop and still worsens p999.
- Guided mode moves min_perf up to max and has the best tail, so a high min_perf is not the cause.

Frame times are from Civ VI on a Deck OLED, with 3 runs per row. Rows 1 to 3 and 4 to 6 are separate sessions.



Guided mode draws more power at light load

Part of the Civ VI run	Active to guided (August 2, n = 3)	Stock to RFC (July 19, n = 7 and 8)
Light load (quietest 10%)	2.78 to 3.51 W (+0.73 W, p = 0.002)	3.17 to 3.33 W (+0.16 W, p = 0.011)
Loading, duration	81.3 to 71.3 s (-12%, p = 0.001)	78.5 to 70.2 s (-11%, p < 0.001)
Loading, mean power	5.87 to 7.07 W (+1.20 W, p < 0.001)	6.02 to 6.77 W (+0.76 W, p < 0.001)
Loading, energy	473 to 499 J (+5.6%, p = 0.044)	472 to 476 J (+0.7%, ns)
Rendering, mean power	12.09 to 12.20 W (+0.11 W, p = 0.056)	12.05 to 12.07 W (+0.02 W, ns)

- Guided mode holds EPP at performance on every CPU, with no knob.
- At light load it draws about a quarter more power. The RFC boosts only busy CPUs.
- The RFC loads the game 11% sooner for the same energy.

Package power is from RAPL on a Deck OLED on the charger. Steady idle and battery operation were not measured.



Related upstream changes in 7.3 and 7.4

- 7.3 makes dynamic EPP a per-policy mode. On battery it reverts to `balance_performance`.
- 7.4 queues per-SoC EPP tables.
- Christian Loehle's patch keeps a uclamp-raised frequency into tickless idle.
- The current tree pins `min_perf` at max on Van Gogh. It is unclear whether this is intended.



Which layer decides when the power cost of a boost is acceptable?

- The boost costs about 1 W while cores are busy.
- It buys about 30% in 1%-low fps.
- A GPU-bound vkmark run shows no cost from the shared TDP.
- The decision could belong to the kernel, the OS or the game.

