

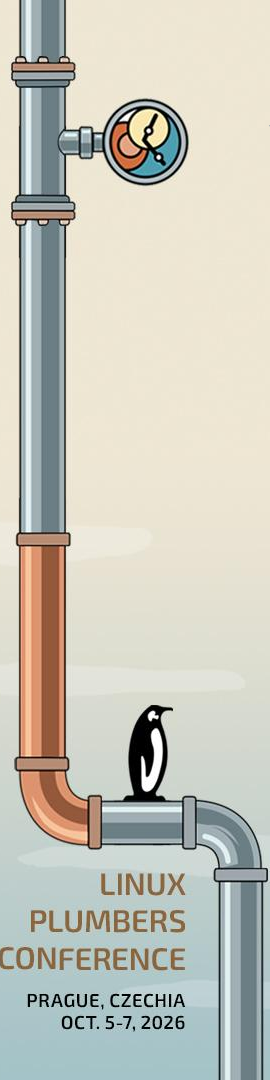
Power capping cooling device

Daniel Lezcano



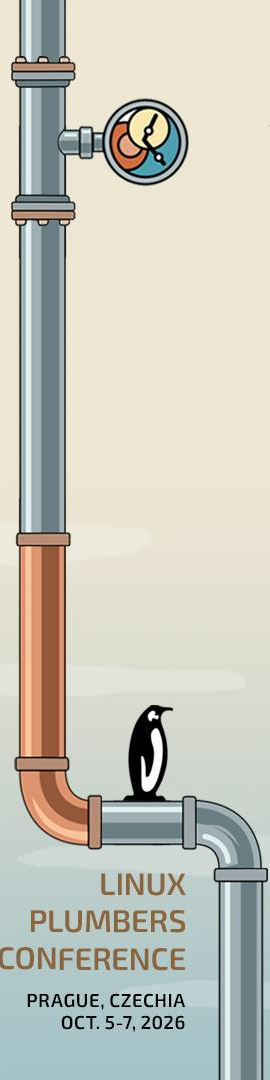
LINUX
PLUMBERS
CONFERENCE

PRAGUE, CZECHIA / OCT. 5-7, 2026



What do we have ? Thermal side

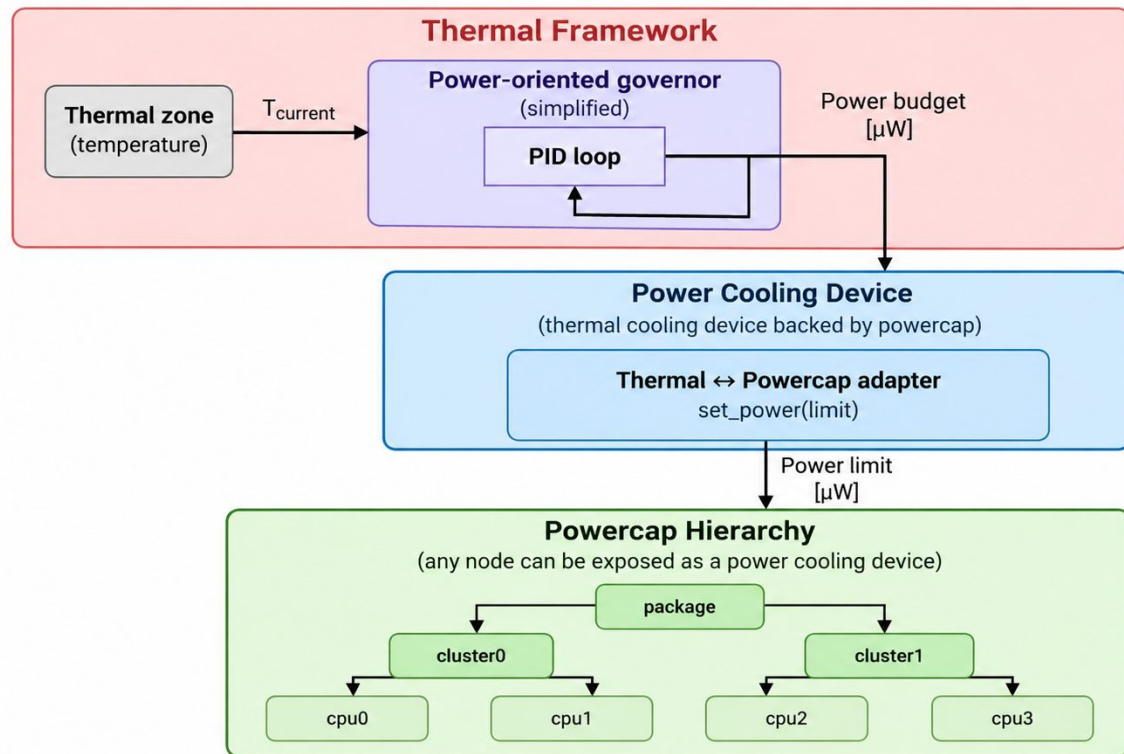
- cpufreq + devfreq cooling with the energy model
 - Working with discrete states
- Intelligent Power Allocator
 - Working with power numbers
- The drivers uses power $\Leftrightarrow$ state conversions

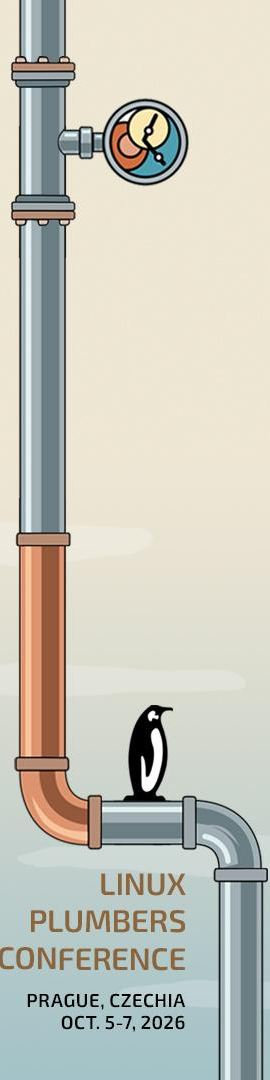


What do we have ? Powercap side

- A generic power capping framework
 - Working with power numbers
- Backend drivers
 - RAPL, DTPM, SPEL (wip), and more coming
- A hierarchy of zones: a power constraint set on a parent node impacts the children nodes

A powercap cooling device



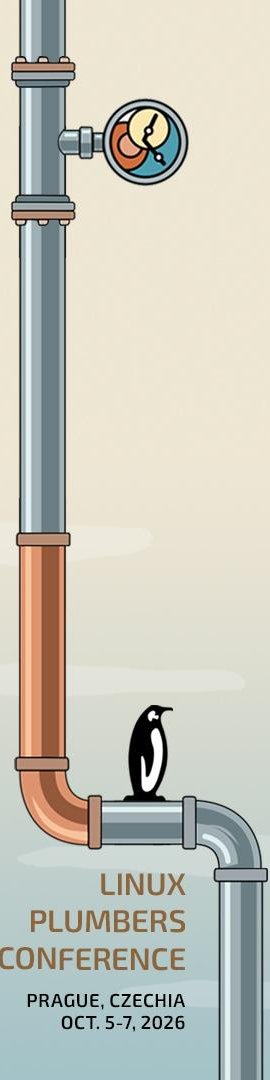


Benefits

- Remove power related code in cpufreq and devfreq
- Hierarchy allows flexible actions from several cooling device
- Multiple constraints from different places can be centralized in the powercap framework
- Allows easy connection between the thermal framework and the powercap framework

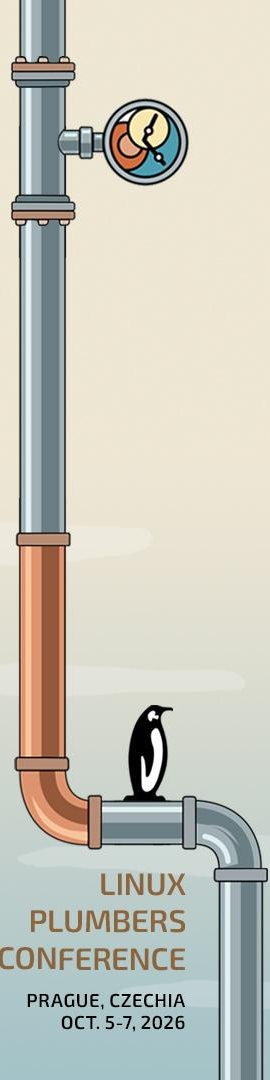
LINUX
PLUMBERS
CONFERENCE

PRAGUE, CZECHIA
OCT. 5-7, 2026



Frictions / questions

- Device tree code
 - A powercap-of framework like thermal-of ?
- Bind a powercap node with a thermal zone
 - What about a virtual powercap node (needed to take full advantage of the hierarchy) ?
- From where shall we register the cooling device node ?
 - Powercap framework ? Or from the backend driver ?



Frictions / questions

- User/sysfs + kernel concurrency
 - Add in-kernel interface, do not use ops->* directly anymore
- Semantic for the IPA
 - Eg. What is max_state for power?
 - New simpler PID power based governor?
- Cpufreq / devfreq vs powercap cooling device conflict
 - Mutual exclusion Kconfig option?