



Contribution ID: 133

Type: **not specified**

## Improving monitoring of power saving states

*Tuesday, 14 November 2023 17:45 (15 minutes)*

ChromeOS is the second most-popular Linux distribution, designed to be long-lasting on battery and snappy, even for low-cost devices. This is achieved by leveraging various power-saving policies on multiple different platforms, including x86 (Intel, AMD), and ARM. Starting at the point of device development, we monitor and verify cpu idle, module (package/device) idle and system deep sleep power state residencies and behaviors. This data allows us to test and tune our system policies (that are applied by our userspace tools) for optimizing power and performance. In addition, we monitor this data through lab and user telemetry, to ensure that existing policies continue to be effective,

Retrieving this data from the Linux kernel has its challenges: standardized interfaces such as cpuidle are often inaccurate due to insufficient hardware visibility, and may suffer from lack of atomicity (when reading residencies in bulk). Reading module idle and system sleep state counters is not standardized, and happens through a combination of debugfs and raw registers, which requires complex userspace code paths depending on SoC vendor, architecture, generation, and even kernel version. Some data is duplicated, and some crucial information is missing, such as the deepest sleep state reached, and time spent there.

In this talk we'd like to discuss the future direction of these kernel interfaces. How can we improve them to meet the needs of modern OS health monitoring?

**Primary author:** VAN ASHBROOK, Sven (Google)

**Co-author:** Mr KARDACH, Stanislaw (Google)

**Presenters:** Mr KARDACH, Stanislaw (Google); VAN ASHBROOK, Sven (Google)

**Session Classification:** Power Management and Thermal Control MC

**Track Classification:** LPC Microconference: Power Management and Thermal Control MC