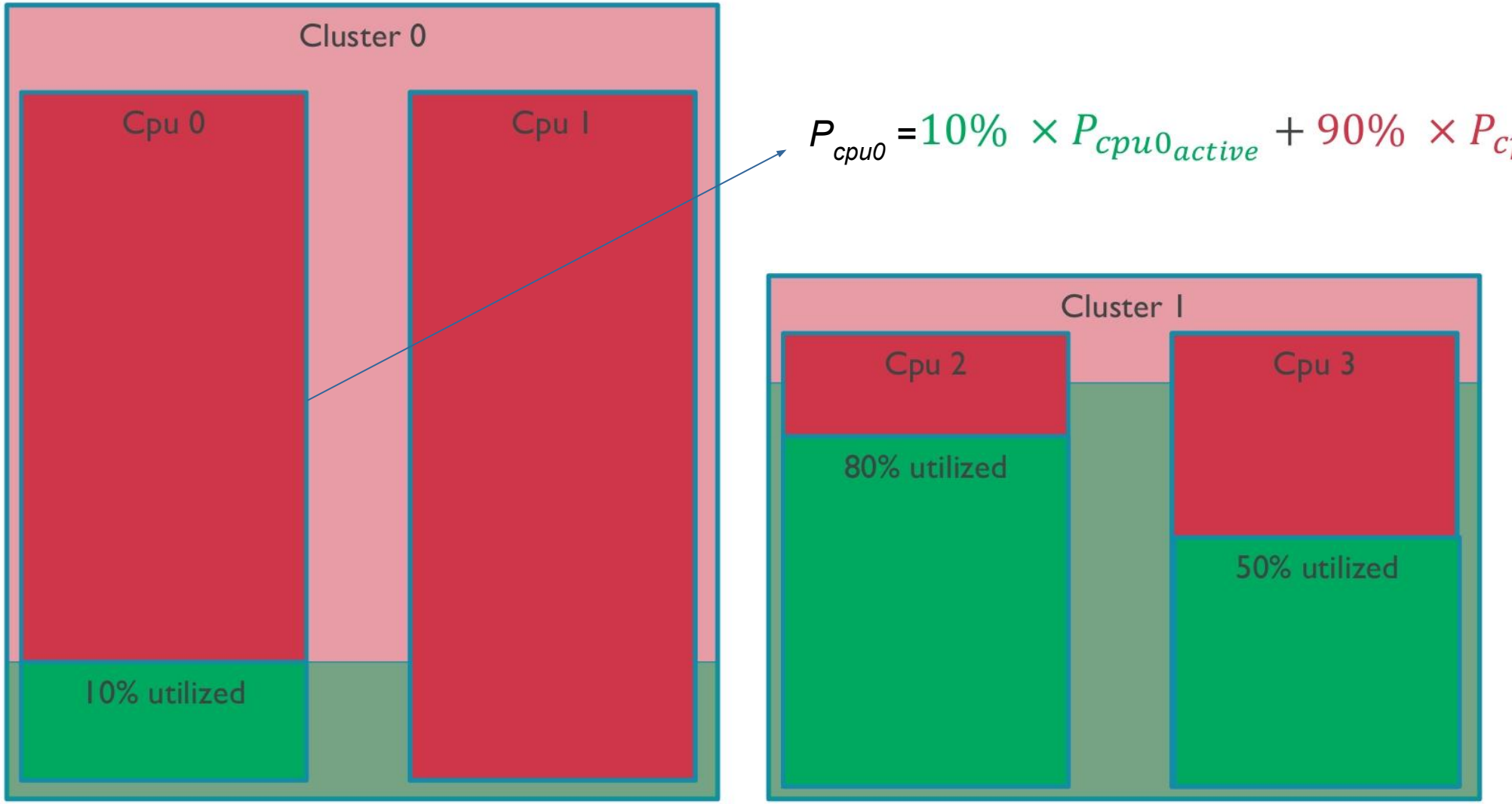
EAS energy model: structure and representation

ARM

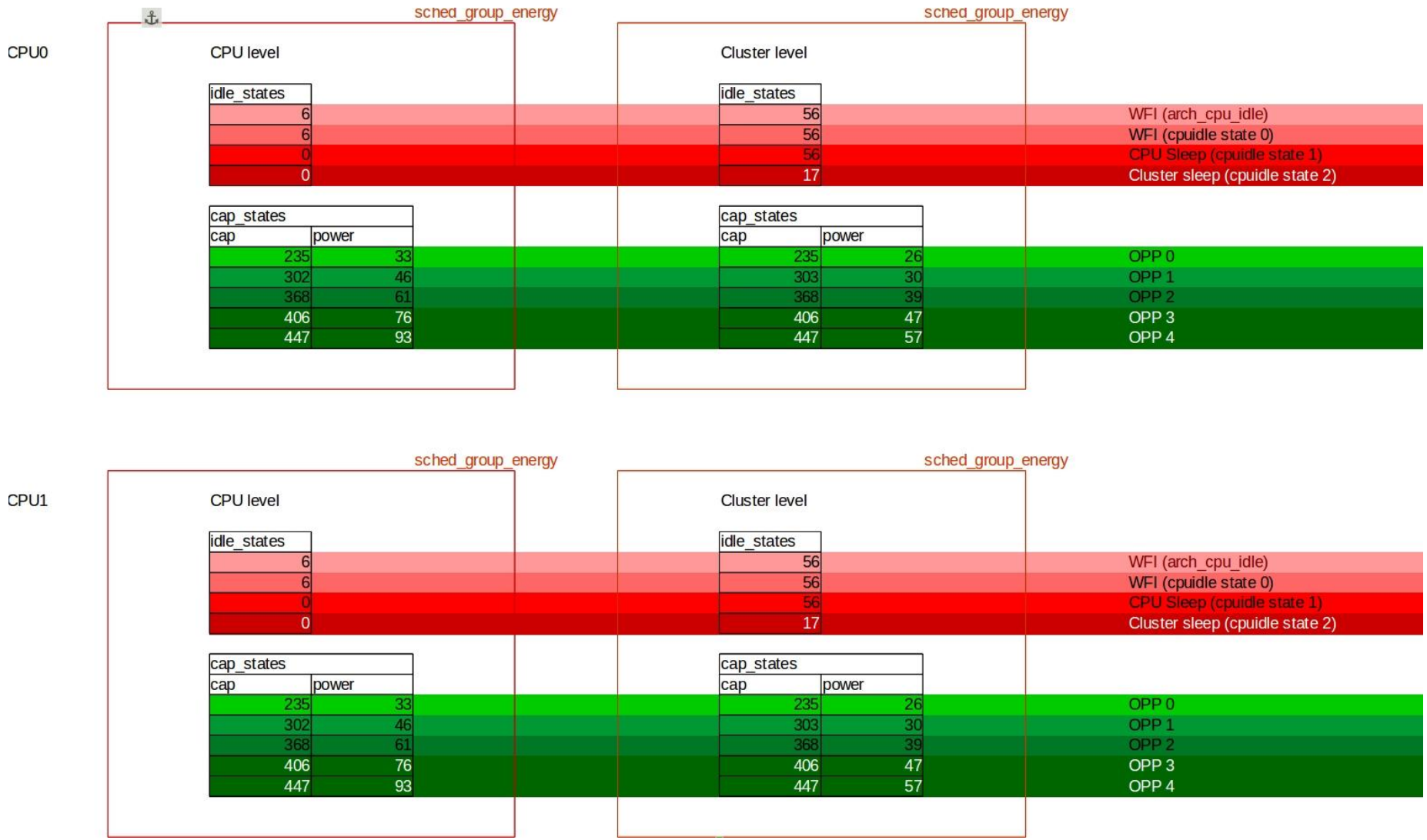
Brendan Jackman

EAS Energy Model - Concept

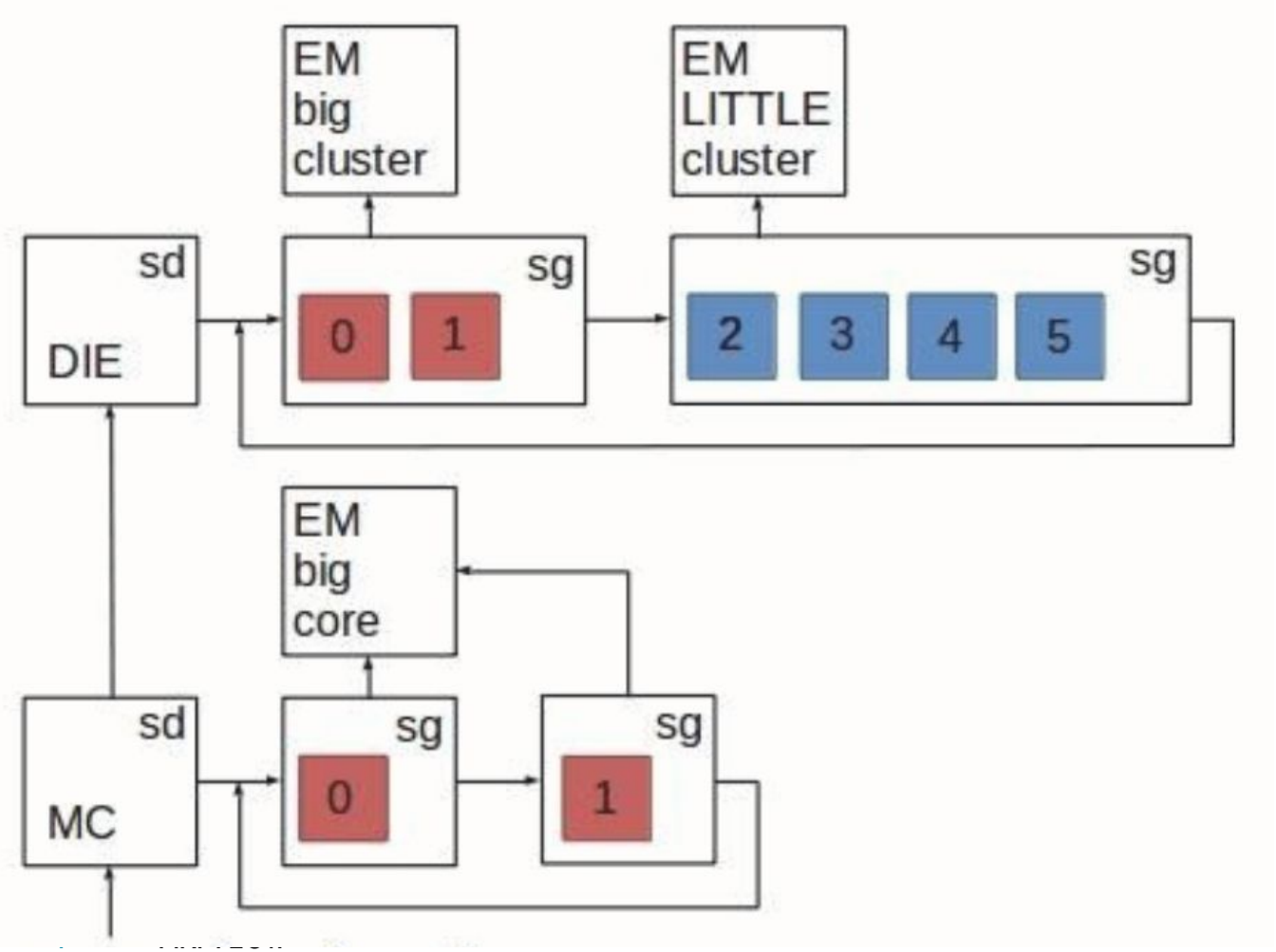


ARM

EAS Energy Model – Data Structure

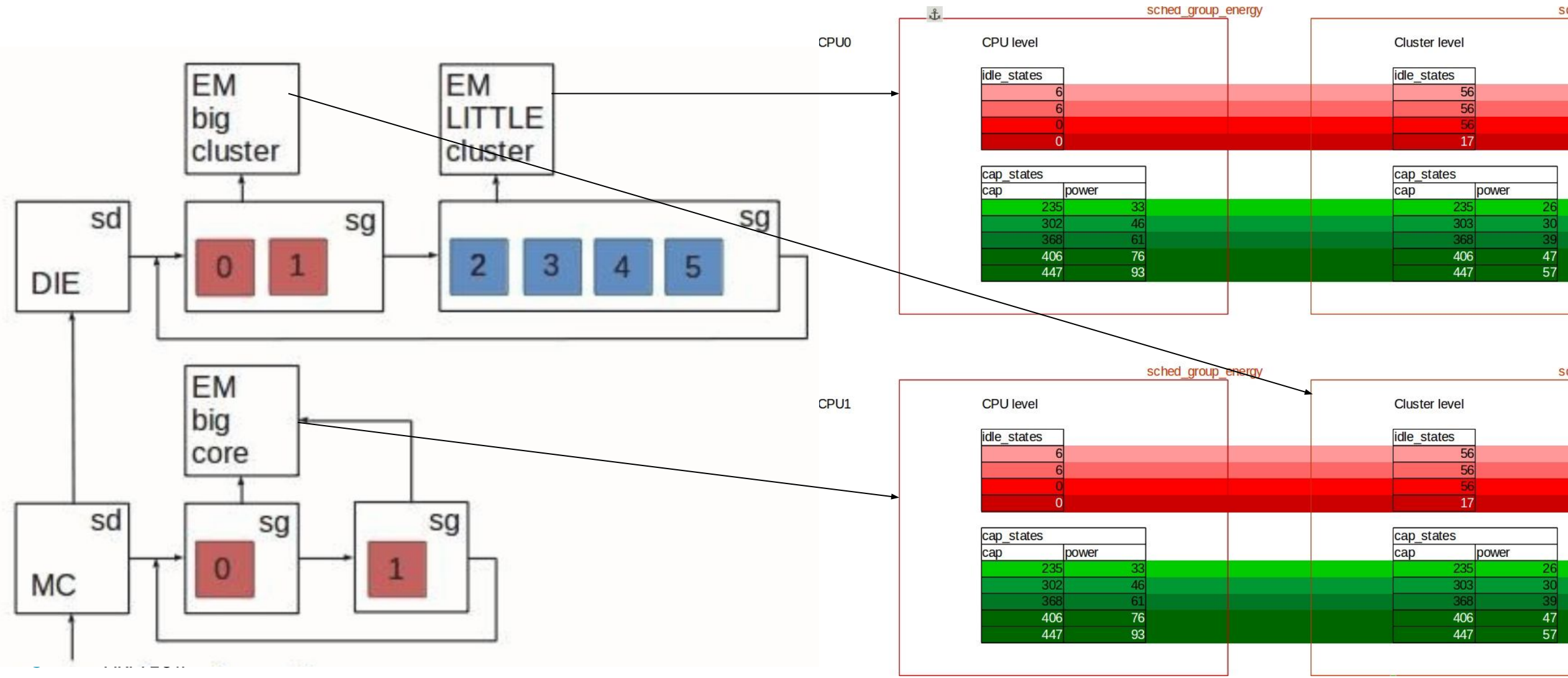


EAS Energy Model – Data Structure



ARM

EAS Energy Model – Data Structure

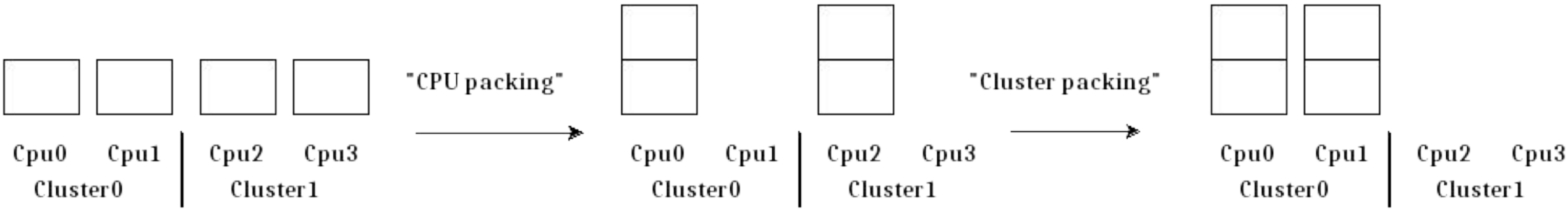


Why Cluster-level Energy Data?

- Should induce “cluster packing”*
- Encourages cluster-level sleep states (switch off shared cache on unused cluster)

- Easy to show effect on scheduler behaviour
- Haven't (yet) clearly demoed energy savings on modern platform.

*"Cluster packing"



Cluster energy - Proposal

- Use `power-domains` hierarchy in DT to express topological (cluster) energy
- Related to work from Lina Iyer & others to get runtime PM to manage OS-initiated (OSI) mode PSCI idle
-

```
A57_0: cpu@0 {
        compatible = "arm,cortex-a57","arm,armv8";
        reg = <0x0 0x0>;
        device_type = "cpu";
        enable-method = "psci";
        next-level-cache = <&A57_L2>;
        clocks = <&scpi_dvfs 0>;
        power-domains = <&A57_0_PD>;
};
```

```
power-domains {
        CLUSTER_A57_PD: cluster-a57-pd {
                #power-domain-cells = <0>;
                domain-idle-states = <&CLUSTER_SLEEP_0>;
        };

        A57_0_PD: a57-pd@0 {
                #power-domain-cells = <0>;
                domain-idle-states = <&CPU_SLEEP_0>;
                power-domains = <&CLUSTER_A57_PD>;
        };

        A57_1_PD: a57-pd@1 {
                #power-domain-cells = <0>;
                domain-idle-states = <&CPU_SLEEP_0>;
                power-domains = <&CLUSTER_A57_PD>;
        };
```

Cluster energy data here

ARM

EAS Energy Model – What we can represent now

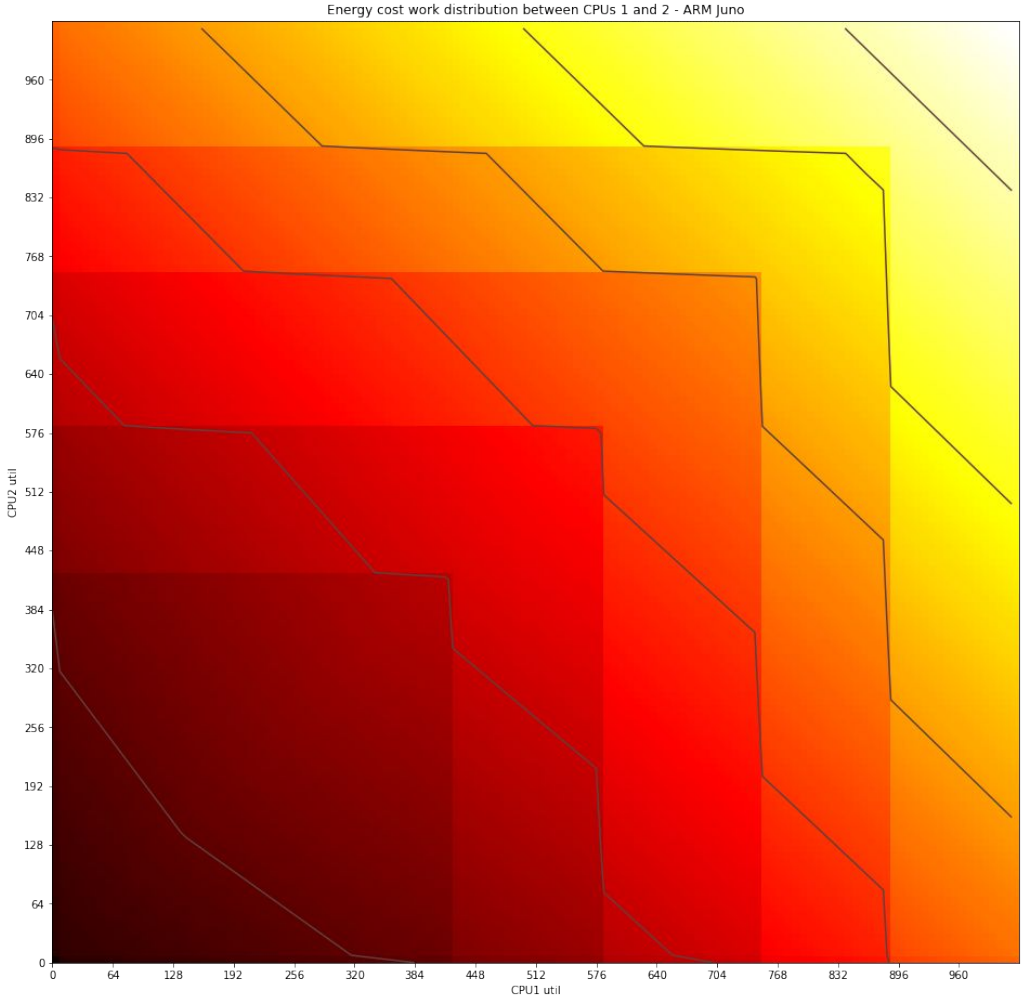
- CPU-level active power can already be derived from:
- `dynamic-power-coefficient` DT property
- `dev_pm_opp` voltage/frequency data
- We can probably do useful stuff using only this data, given:
- Haven't (yet) proved benefit of cluster energy data
- Idle energy has small contribution & is inaccurate anyway

Notes

- “Product” EAS (e.g. Google Pixel kernel, AOSP) has all EAS EM data under a single node
- Can already derive CPU-level active power from DT with `dynamic-power-coefficient` DT property and `dev_pm_opp` voltage/frequency data

Backup >>>

EM heatmaps



EM heatmaps