
KubeCon
CloudNativeCon
Europe 2023
TiKV

Evolution of on-Node Adaptive Power Tuning

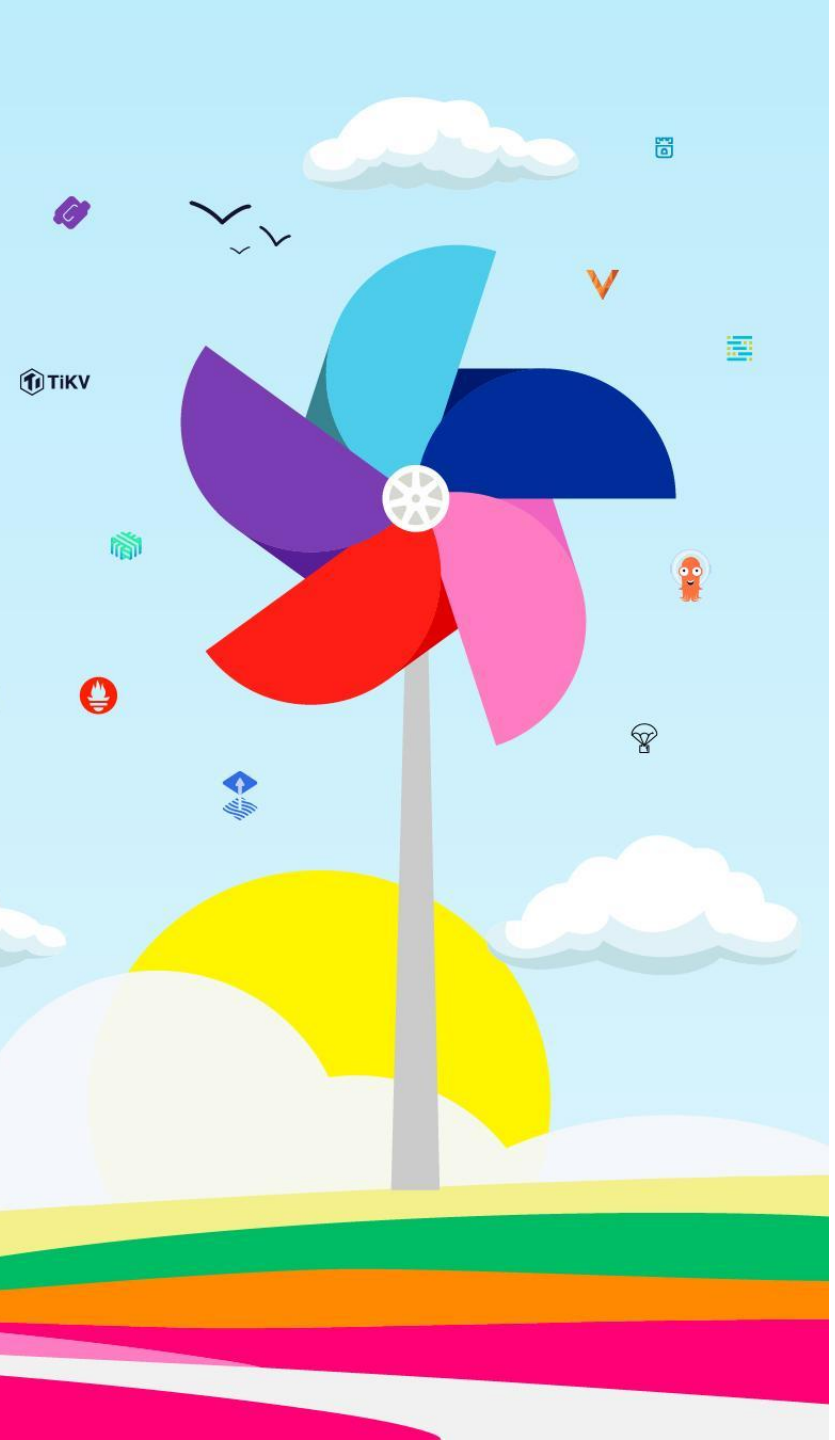
Who are we?

KubeCon CloudNativeCon Europe 2023

Dr. Atanas Atanasov
Senior Cloud-Native Dev.
Intel
atanas.atanasov@intel.com

Rimma Iontel
Chief Architect
Red Hat
riontel@redhat.com

Red Hat | intel



KubeCon | CloudNativeCon
Europe 2023

Power Controls

Node-Level Optimizations

- **Hardware optimization**
  - Granular control of unused components
    - Cores, NICs, disks, hardware accelerators, etc.
  - Turning off unused components
    - NIC, disks, cores, hardware accelerators, etc.
  - Alternate hardware architectures
    - SmartNICs, IPUs, ARM CPUs, etc.
- **OS optimizations**
  - Power efficient profiles
    - Tuned, Performance Profiles
  - Granular selection of settings for energy efficiency
    - CPU governors, p/c-states, uncore frequencies

TuneD

TuneD is a service that monitors the system and optimizes its performance based on ***use case*** specific ***profiles***

- High throughput
- Low latency
- **Power Savings**

- throughput-performance
- latency-performance
- network-latency
- network-throughput
- **balanced**
- **powersave**

CPU Power Controls

P-States: voltage-frequency control states

- CPUfreq in the Linux kernel enables scaling the CPU frequency
- Governor controls allowed settings

```
# cpupower frequency-info
analyzing CPU 0:
  driver: intel_cpufreq
  hardware limits: 800 MHz - 3.50 GHz
  available cpufreq governors: conservative
ondemand userspace powersave performance
schedutil
…
```

C-States: CPU idle sleep states where the processor clock is inactive

- Partially deactivates unused CPUs
- Range: C0 (active) to Cn
- Deeper C-state -> exit latency duration becomes longer

```
# cpupower idle-info

CPUidle driver: intel_idle
CPUidle governor: menu
analyzing CPU 0:
Number of idle states: 4
Available idle states: POLL C1 C1E C6
…
```

Uncore Frequency

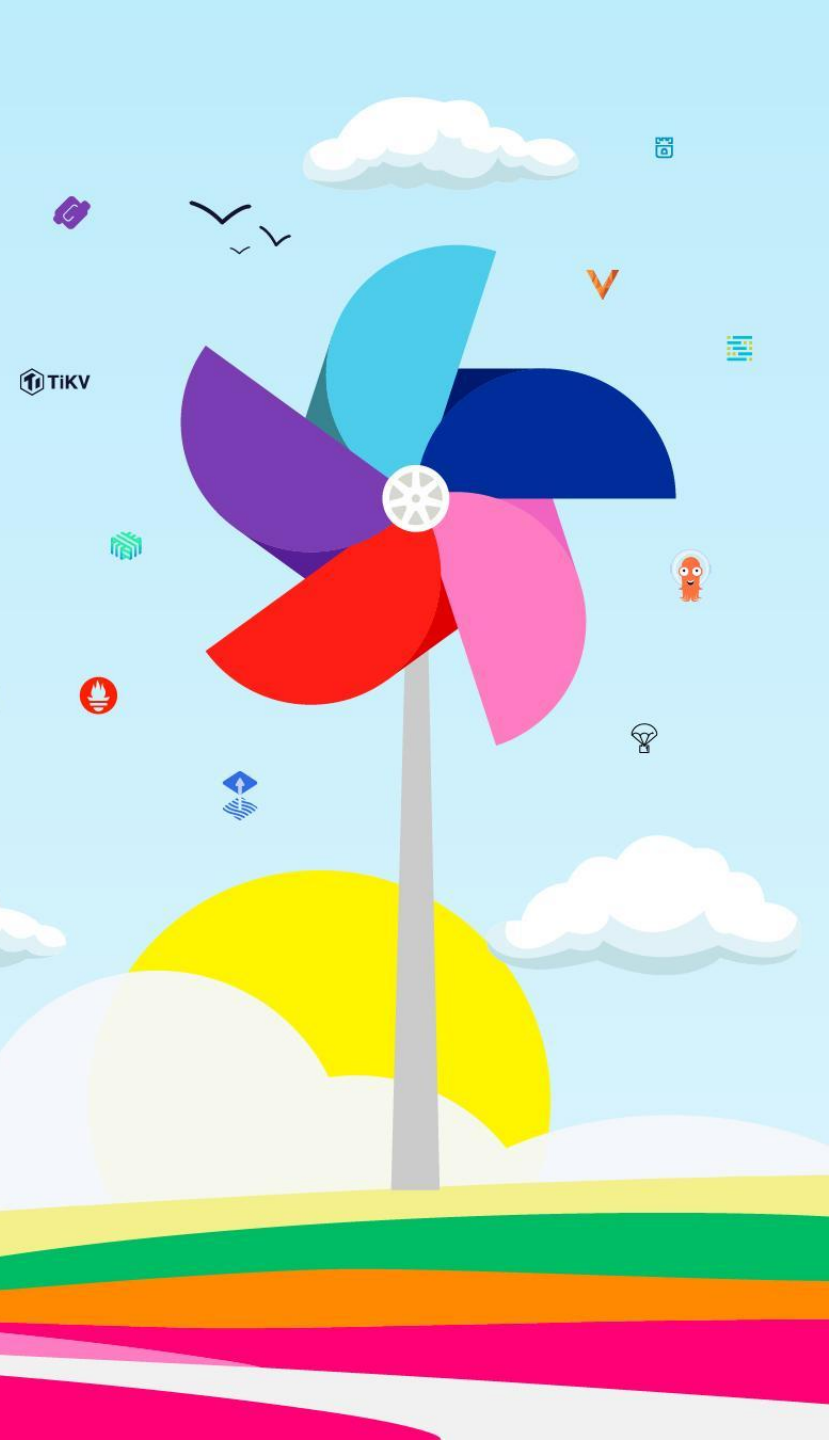
Uncore: components of a processor that are not directly involved in the execution of instructions

- Memory controller
- Cache coherency logic
- Interconnects between processor cores (QPI)

Uncore Frequency Scaling (UFS): Cores interconnect and L3 shared cache frequency scaling for energy efficiency

- BIOS Setting
- Future functionality: OS controls with TuneD or Intel's Kubernetes Power Manager[1]

[1] https://github.com/intel/kubernetes-power-manager



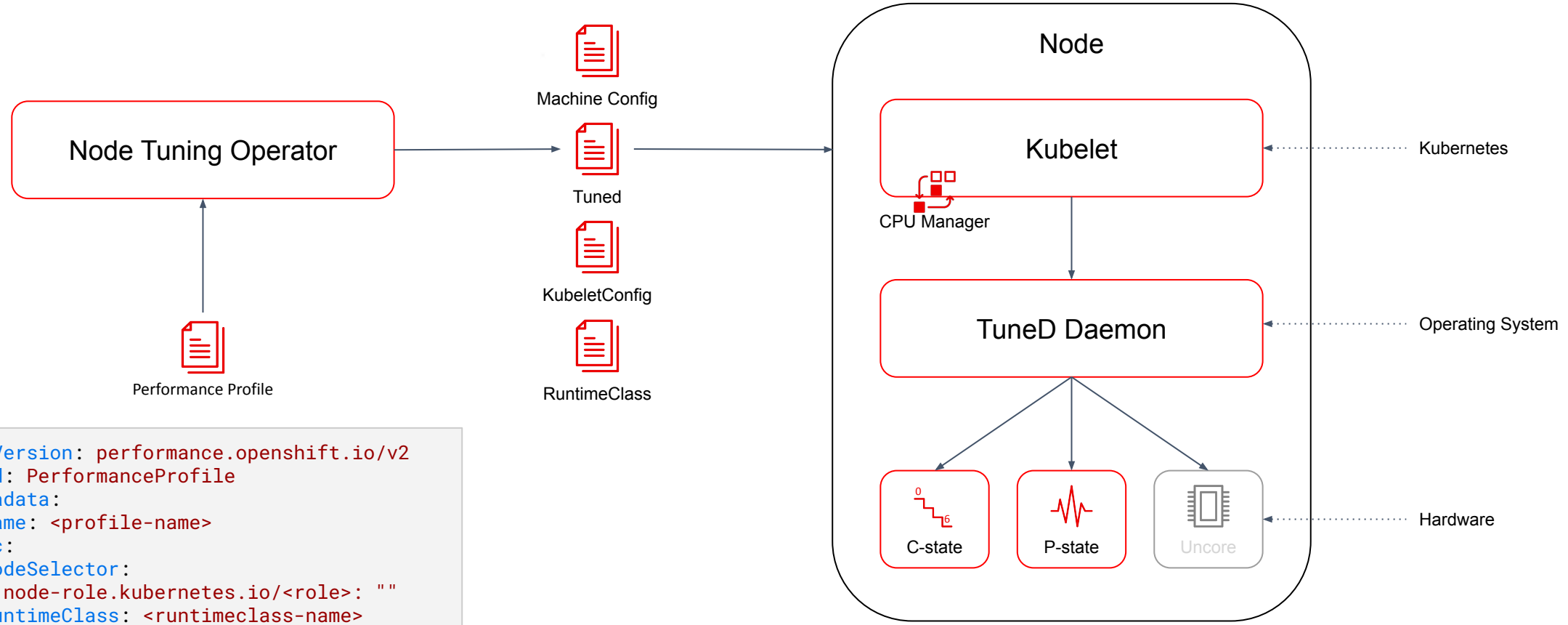
KubeCon | CloudNativeCon
Europe 2023

Kubernetes Power Configuration

Kubernetes Components

- Kubelet
  - **CPU manager** (cpu pinning and isolation)
  - Topology manager (NUMA affinity)
  - Memory manager (Hugepages NUMA affinity)
- Cluster Node Tuning Operator (NTO)
  - Maintains tuning rules for the distributed OCP environment
  - Runs **TuneD** that applies the tuning rules to every node
- Machine Config Operator (MCO)
  - Abstracts operating system and CRI-O configuration changes
- Performance Profile Controller
  - Computes tuning rules from the provided high level description
  - Provides configuration snippets to all lower level components

Under the Hood




```yaml
apiVersion: performance.openshift.io/v2
kind: PerformanceProfile
metadata:
  name: <profile-name>
spec:
  nodeSelector:
    node-role.kubernetes.io/<role>: ""
  runtimeClass: <runtimeclass-name>
  tuned: <namespace>/<tuned-profile-name>
…
```

Node Tuning Operator

Node Tuning Operator energy optimizations:

- **CPU core** disablement/offline
- **CPU Governor** selector per core
- **CPU Frequency** tuning for group of cores
- Granular **power optimizations** for mixed workloads

Permanently Offline CPUs

Use case: the worker nodes of the cluster have been deployed with extra CPU capacity that will be used in the future. How to turn them off until we need them?

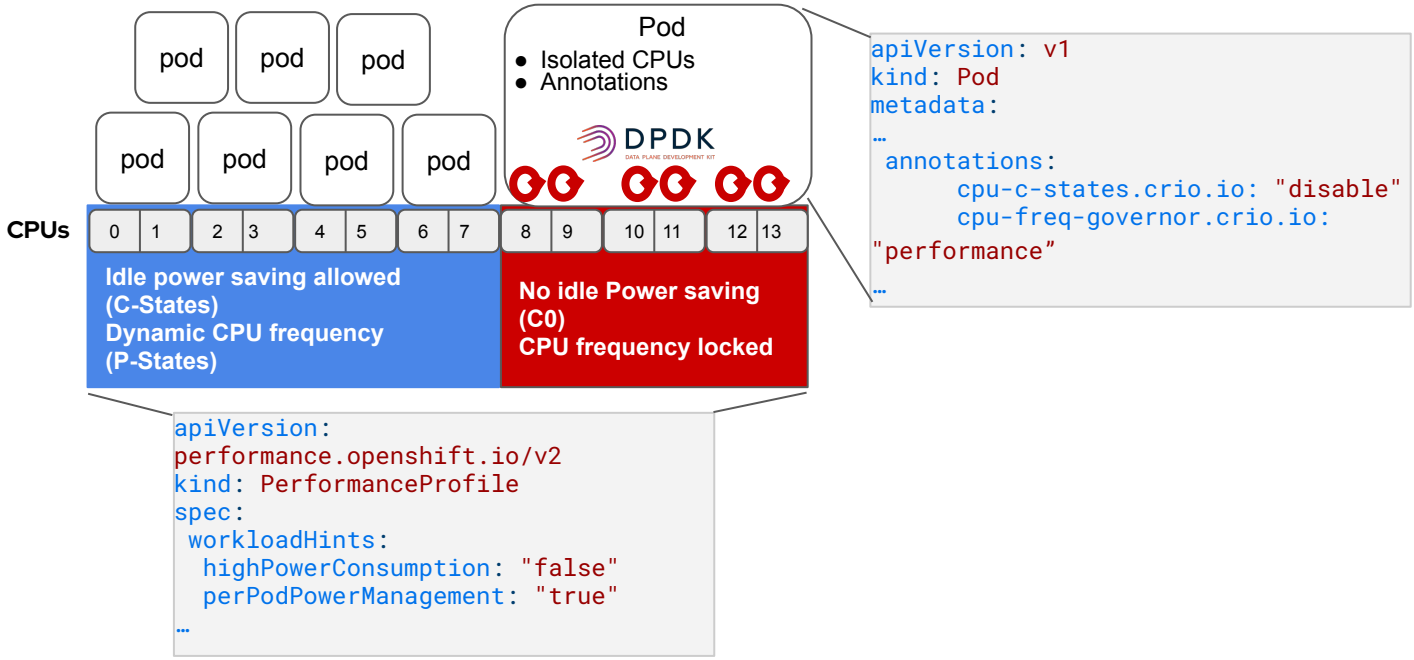
- Shut down select CPUs with PerformanceProfile
- Done at boot time, so any configuration change requires a reboot
  - Sets maxcpus=X in kernel boot arguments
  - Changes /sys/devices/system/cpu/cpuX/online

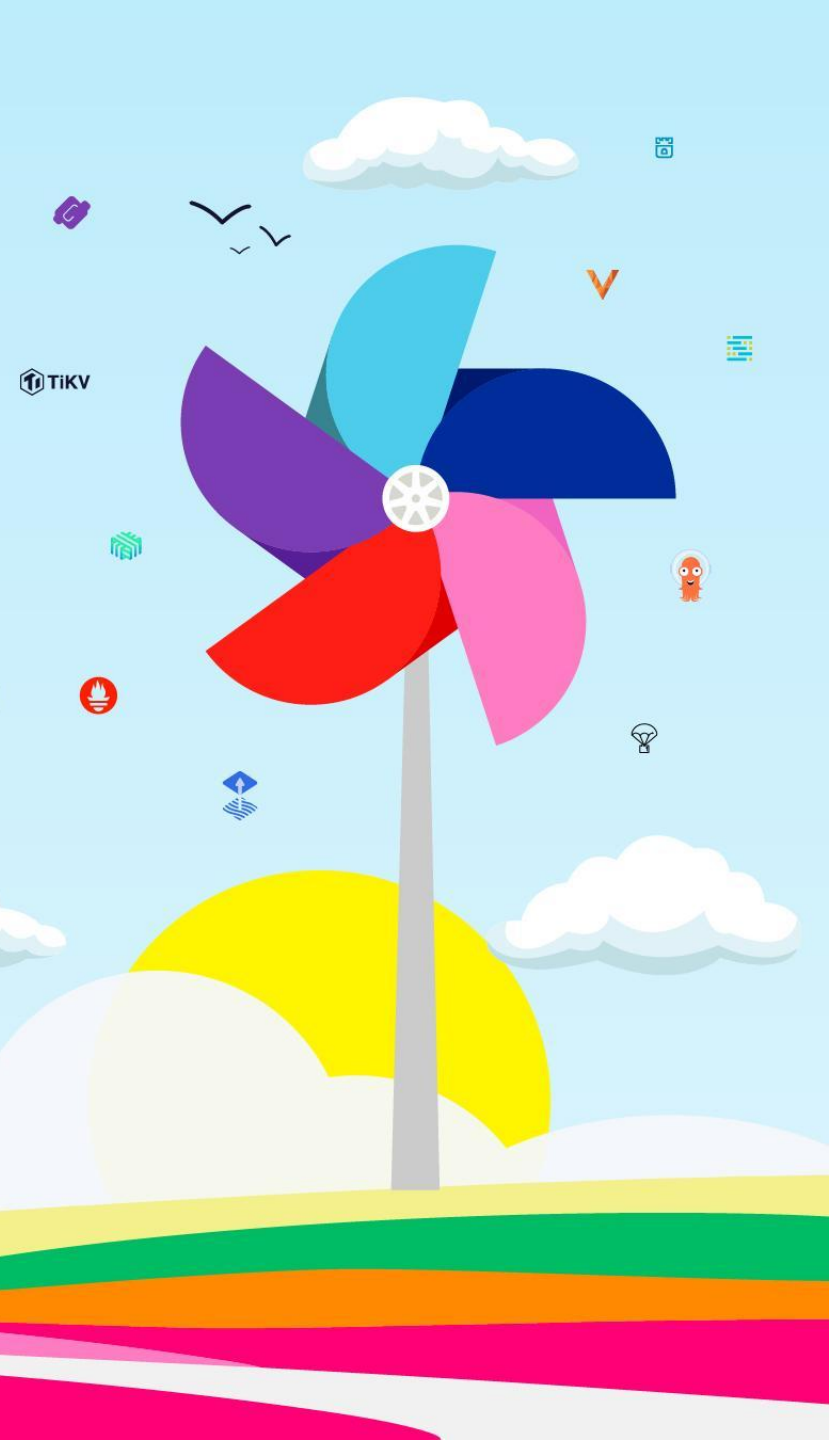
```
apiVersion:
performance.openshift.io/v2
kind: PerformanceProfile
metadata:
 name: <profile-name>
spec:
 cpu:
  isolated: "2-21,26-37"
  reserved: "0-1,24-25"
  offlined: "38-42"
…
```

Power optimizations for mixed workloads

Configure CPU Power states **per pod** to support mixed workloads

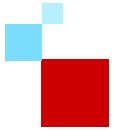
- Enable CPU power savings features on all CPUs by default
- Apply performance optimizations per pod through annotations





KubeCon | CloudNativeCon
Europe 2023

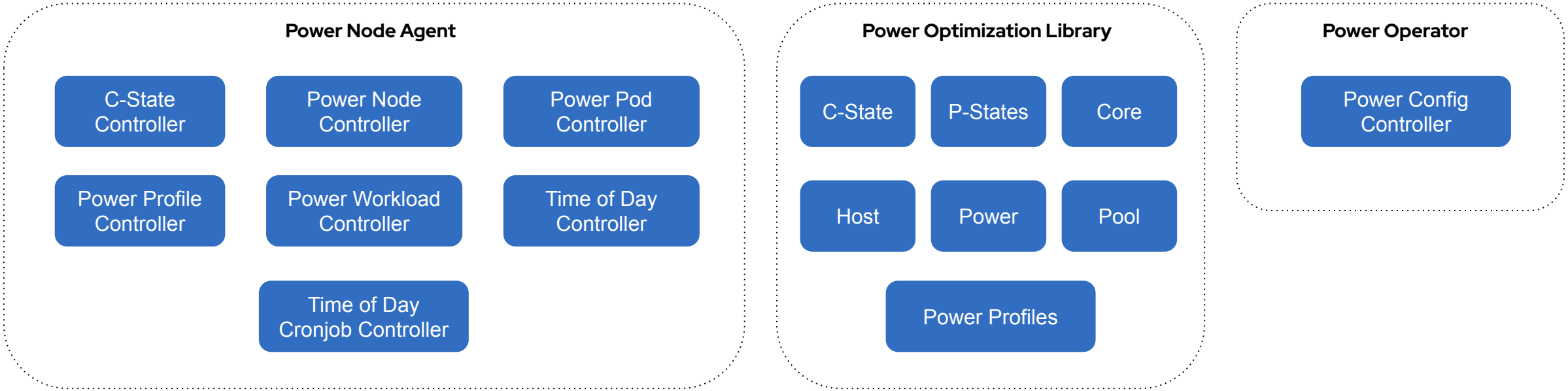
Kubernetes Power Manager



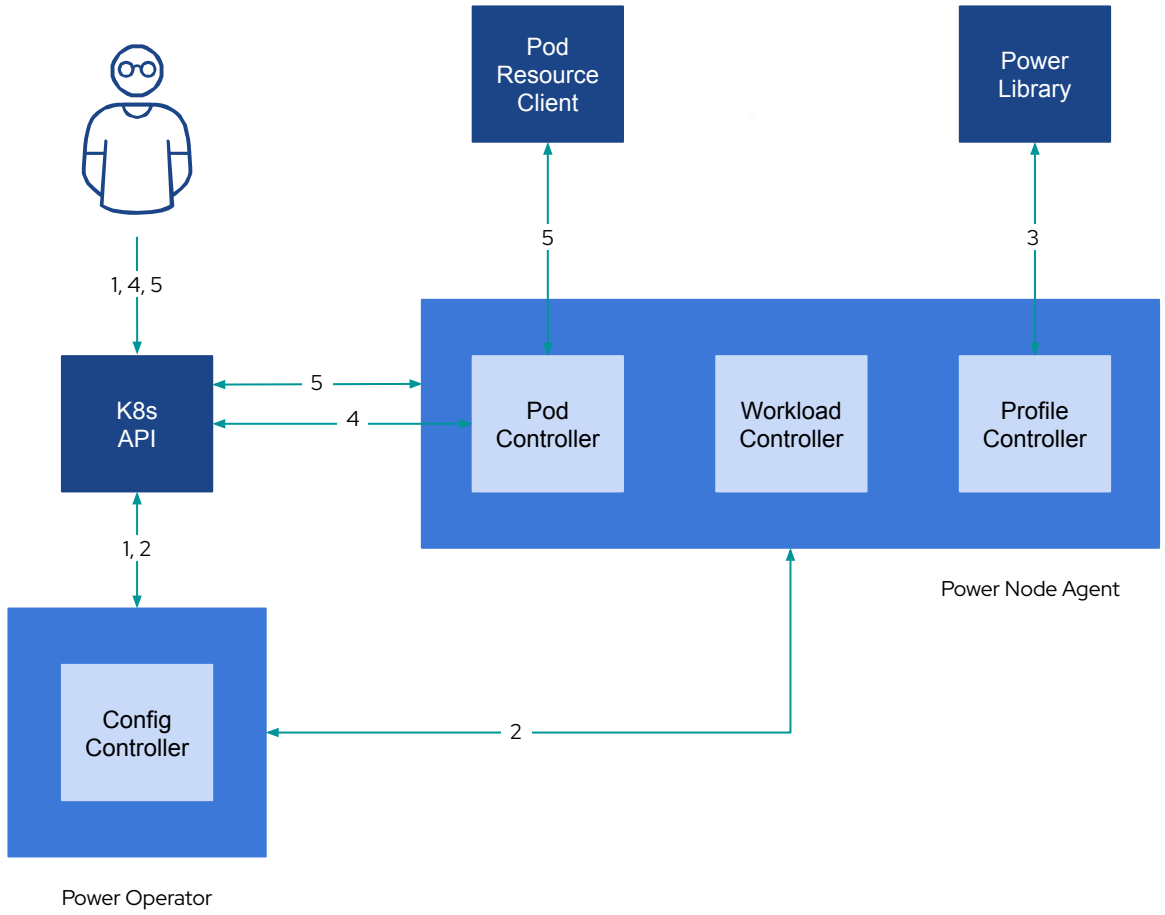
- Reduce operational overheads
- Lower power consumption by controlling the frequencies of the shared pool cores
- Turn off uncore functionality, as needed
- Choose specific governors
- Play with various sleep states

Benefits of the K8s Power Manager to our Ecosystem

K8s Power Manager Component Diagram



K8s Power Manager Flow Diagram



1. User creates Config custom resource
2. Config controller creates Power Node Agent / Profiles
3. Profiles / Pools created in Power Optimization Library
4. Shared Profile / Workload created by the user
5. User creates Pod requesting Profile. Power Node Agent configures the associated cores

Example of Power Profile Spec

Power Profile Spec

```
apiVersion:"power.intel.com/v1"
Kind: Power Profile
Metadata:
  name: performance
Spec:
  name: "Performance"
  max: 3200
  min: 2800
  epp: "performance"
```

Power Profile Spec

```
apiVersion:"power.intel.com/v1"
Kind: Power Profile
Metadata:
  name: balance-performance
Spec:
  name: "Balance-Performance"
  max: 3000
  min: 2500
  epp: "balance-performance"
```

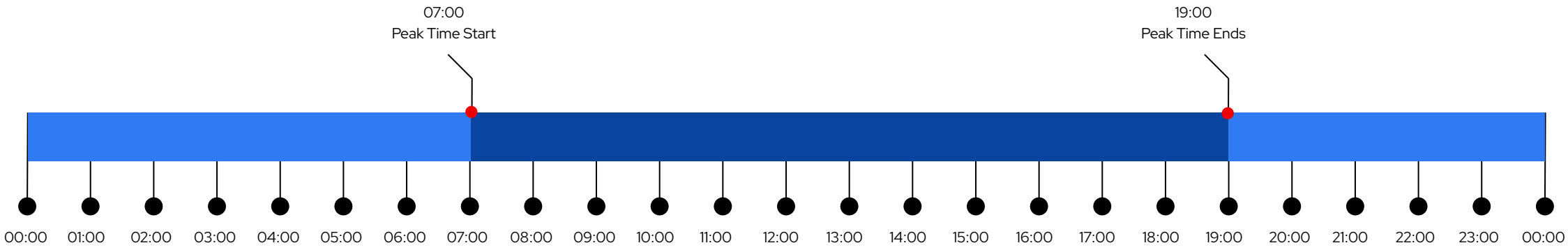
Power Profile Spec

```
apiVersion:"power.intel.com/v1"
Kind: Power Profile
Metadata:
  name: power
Spec:
  name: "Power"
  max: 1500
  min: 1000
  epp: "power"
```

Power Profile Spec

```
apiVersion:"power.intel.com/v1"
Kind: Power Profile
Metadata:
  name: balance-power
Spec:
  name: "Balance-Power"
  max: 2200
  min: 1800
  epp: "balance-power"
```

Time of Day

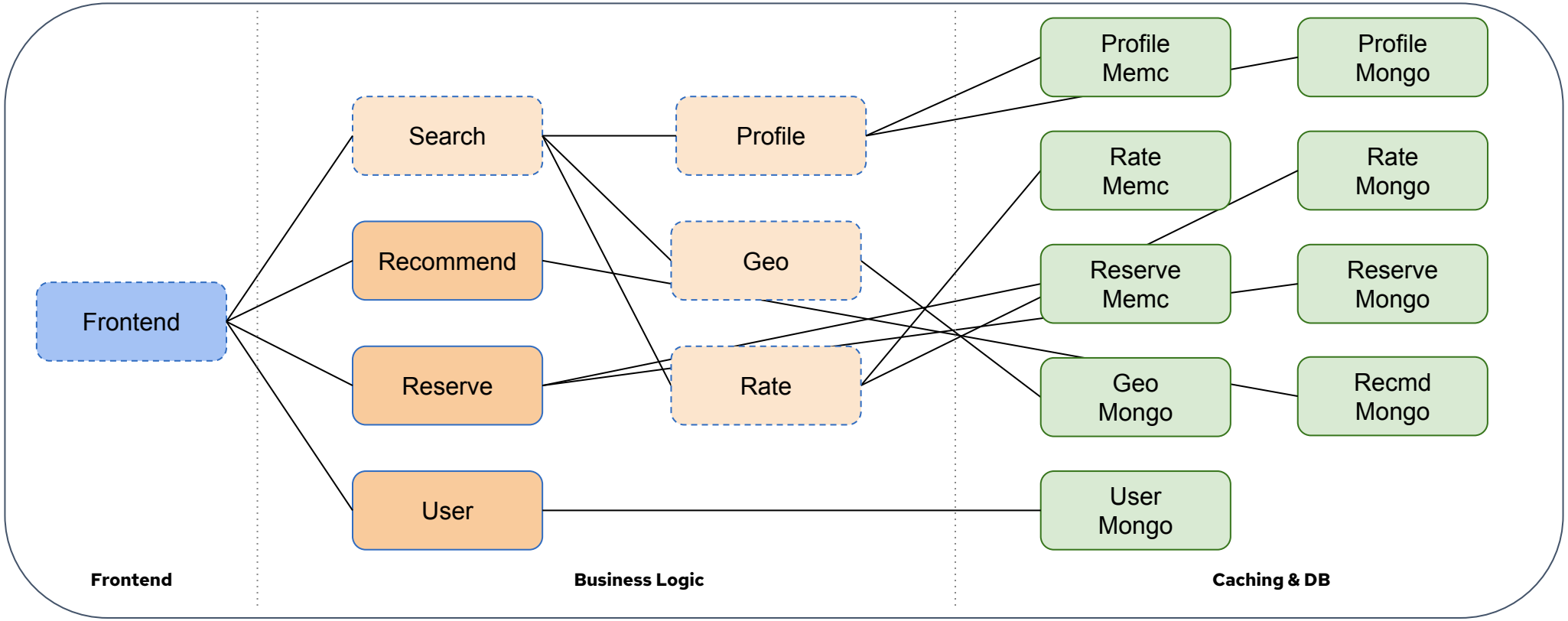


Sleep Time

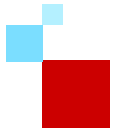
Active Time

```
apiVersion: time-of-day/v1
kind: peak-time
metadata:
  name: peak-time
spec:
  weekdays: "1-5"
  sleep: "19:00"
  active: "07:00"
  timeZone: "Europe/Ireland"
```

Use Case – DSB Hotel Reservation

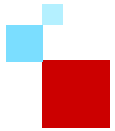


- DSB(DeathStarBench) includes representative microservices workloads open sourced by Cornell University
  - Hotel Reservation, Social Network Service, Media Service
  - Workloads to simulate real world services
- Wrk is used as load generator, traffic specified via lua script
- Created larger dataset with 10M users/properties
- Uses gRPC for communication between services



Run Power Manager in OCP

1. Configure TuneD
2. Deploy Power Manager
3. Deploy Power Profiles
4. Deploy Power Workload



1. TuneD Profile
2. Deploy Power Manager

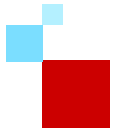
```yaml
apiVersion: tuned.openshift.io/v1
kind: Tuned
metadata:
  name: intel-kpm-hotfixes
  namespace: openshift-cluster-node-tuning-operator
spec:
  profile:
  - data: |
      [main]
      summary=Configuration changes profile inherited from performance created tuned

      include=performance
      # ensure intel_pstate is loaded and in active mode
      [bootloader]
      cmdline_removeKernelArgs=-intel_pstate=disable -intel_pstate=no_hwp -intel_pstate=passive

      [cpu]
      # disabled as it clashes with power manager
      enabled=false

      # ensure required modules are loaded,
      # no module options
      [modules]
      intel_cstate=
    name: openshift-intel-kpm-hotfixes
  recommend:
  - machineConfigLabels:
      machineconfiguration.openshift.io/role: intel-kpm
    priority: 15
    profile: openshift-intel-kpm-hotfixes
```

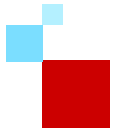
https://github.com/intel/kubernetes-power-manager



3. Deploy Power Profiles

```yaml
apiVersion: "power.intel.com/v1"
kind: PowerProfile
metadata:
   name: shared-example-node1
spec:
   name: "shared-example-node1"
   max: 1500
   min: 1000
   epp: "power"
   governor: "powersave"
```

```yaml
apiVersion: "power.intel.com/v1"
kind: PowerProfile
metadata:
   name: performance-example-node
spec:
   name: "performance-example-node"
   max: 3700
   min: 3300
   epp: "performance"
   governor: "performance"
```



4. Deploy Workload

performance.yaml

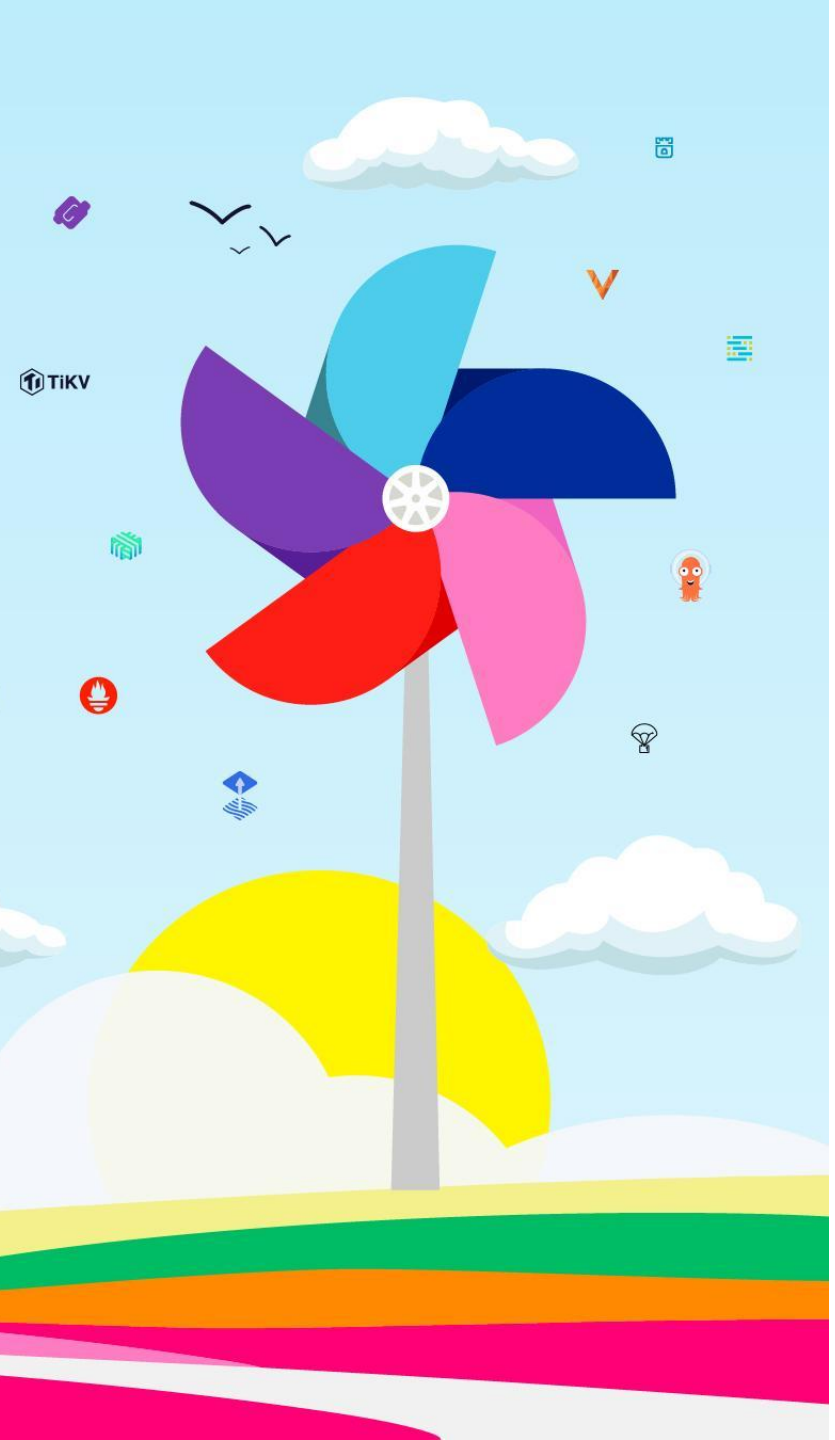
root > demo > ! performance.yaml

```yaml
apiVersion: v1
kind: Pod
metadata:
  name: performance-pod
spec:
  containers:
  - name: performance-container
    image: ubuntu
    command: ["/bin/sh"]
    args: ["-c", "sleep 15000"]
    resources:
      requests:
        memory: "200Mi"
        cpu: "2"
        power.intel.com/performance: "2"
      limits:
        memory: "200Mi"
        cpu: "2"
        power.intel.com/performance: "2"
```

balance-power.yaml

root > demo > ! balance-power.yaml

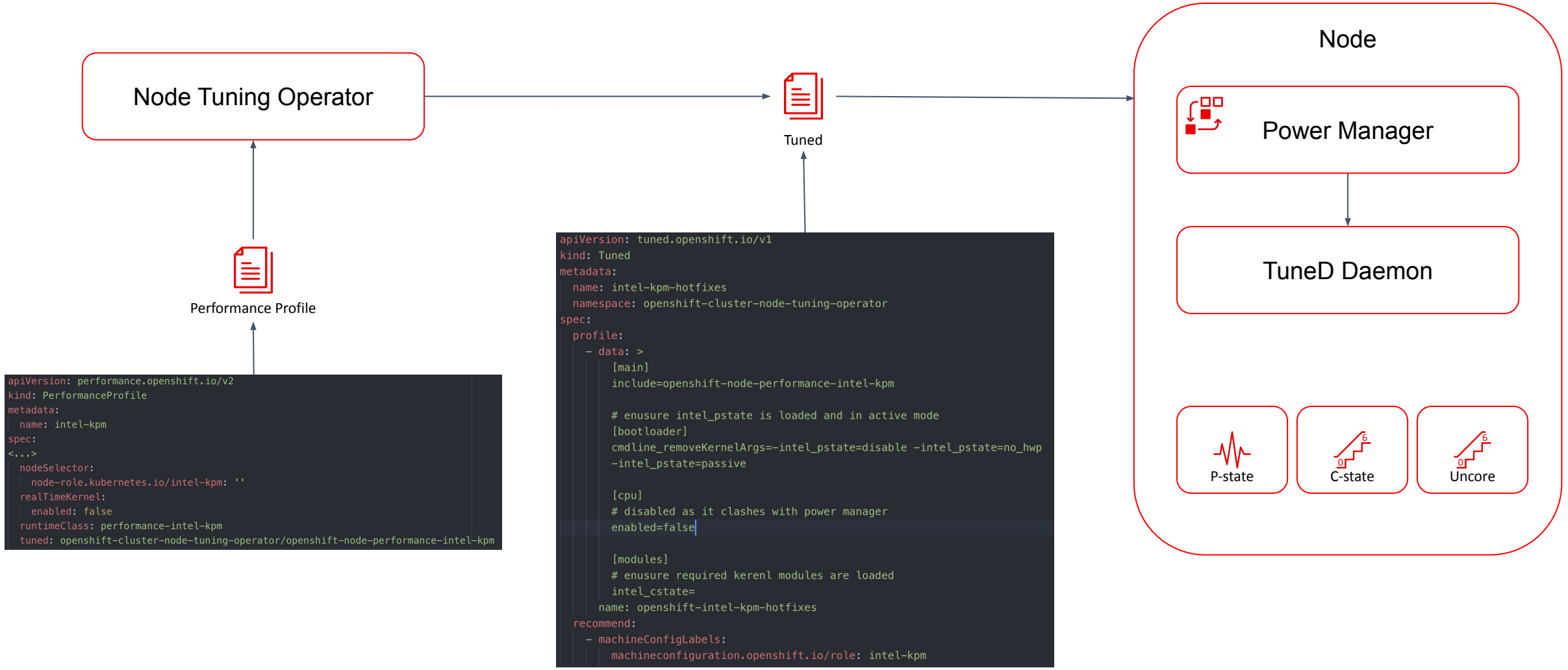
```yaml
apiVersion: v1
kind: Pod
metadata:
  name: balance-power-pod
spec:
  containers:
  - name: balance-power-container
    image: ubuntu
    command: ["/bin/sh"]
    args: ["-c", "sleep 15000"]
    resources:
      requests:
        memory: "200Mi"
        cpu: "2"
        power.intel.com/balance-power: "2"
      limits:
        memory: "200Mi"
        cpu: "2"
        power.intel.com/balance-power: "2"
```



Demo Time!

KubeCon | CloudNativeCon
Europe 2023

Please scan the QR Code above
to leave feedback on this session




```yaml
apiVersion: performance.openshift.io/v2
kind: PerformanceProfile
metadata:
  name: intel-kpm
spec:
<...>
  nodeSelector:
    node-role.kubernetes.io/intel-kpm: ''
  realTimeKernel:
    enabled: false
  runtimeClass: performance-intel-kpm
  tuned: openshift-cluster-node-tuning-operator/openshift-node-performance-intel-kpm
```

```yaml
apiVersion: tuned.openshift.io/v1
kind: Tuned
metadata:
  name: intel-kpm-hotfixes
  namespace: openshift-cluster-node-tuning-operator
spec:
  profile:
    - data: >
        [main]
        include=openshift-node-performance-intel-kpm

        # enusure intel_pstate is loaded and in active mode
        [bootloader]
        cmdline_removeKernelArgs=-intel_pstate=disable -intel_pstate=no_hwp
        -intel_pstate=passive

        [cpu]
        # disabled as it clashes with power manager
        enabled=false

        [modules]
        # enusure required kerenl modules are loaded
        intel_cstate=
      name: openshift-intel-kpm-hotfixes
  recommend:
    - machineConfigLabels:
        machineconfiguration.openshift.io/role: intel-kpm
```