KubeCon Europe 2022 – Valencia – May 18

Working your Cluster: Smarter Scheduling Decisions for Your Workloads

Denisio Togashi, Madalina Lazar

Agenda

- Resource States & Scheduling
- Telemetry Aware Scheduler
  - *Intro*
  - *Workflow*
  - *Scheduling policies*
- Demo
- Q&A

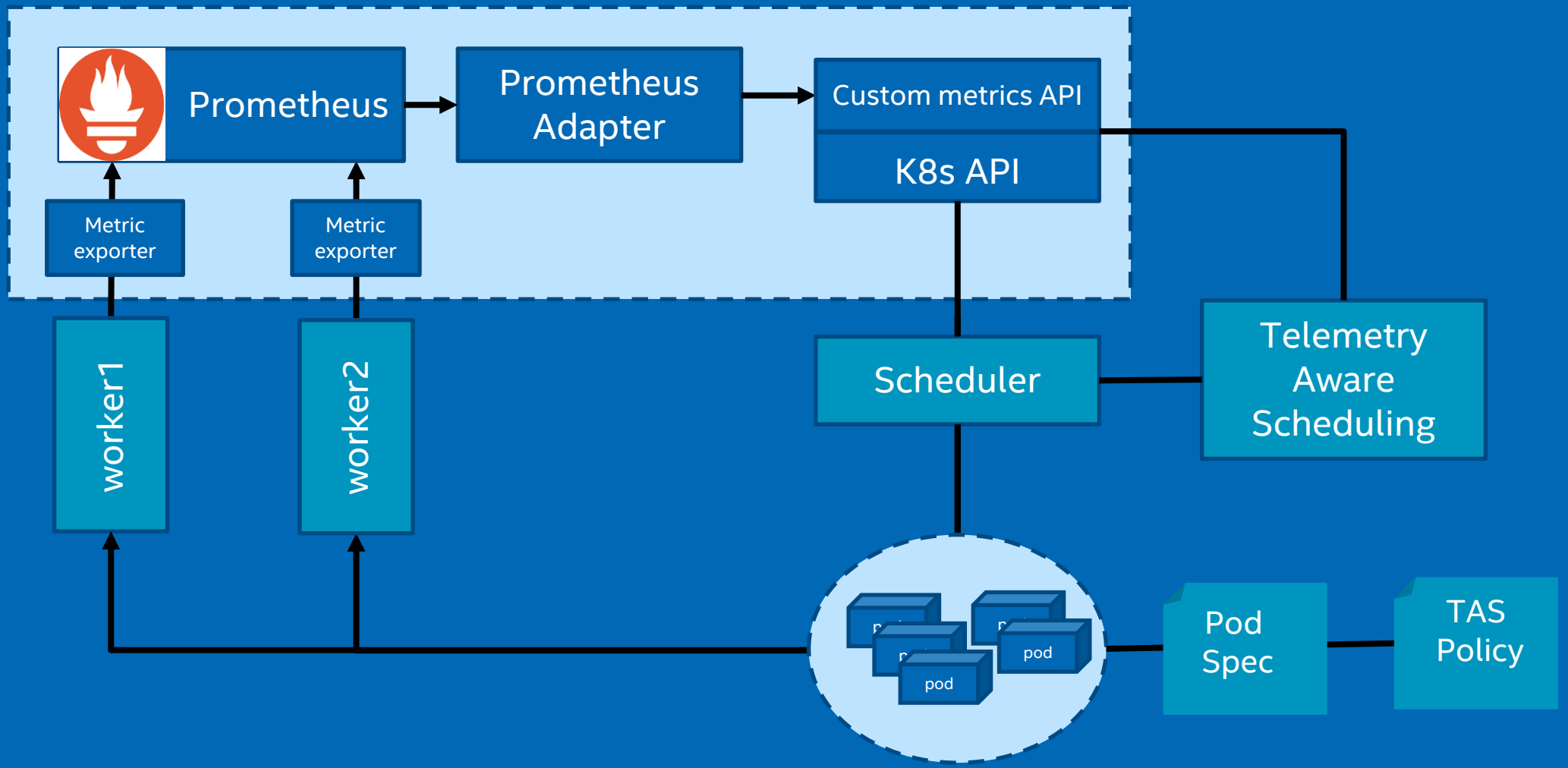
Resource States & Scheduling

- Avoid scheduling to unhealthy nodes.
- Migrate pods away from an unhealthy node.
- When scheduling, consider node properties, such as temperature, current load, power, etc.
- Give support to external components.

Telemetry Aware Scheduling - *Intro*

- **Extender** of the default K8s Scheduler
- **Uses telemetry data** to make scheduling and de-scheduling decisions in Kubernetes.
- **Uses policies to enable rule-based decisions on pod placement.** The rules are powered by metrics collected from nodes.
- Knows how to **interpret filtering & scoring and utilizes node affinity rules**.
- **The policies support multi-metric rules.** These can be built by joining different metric combinations and logical operators like “anyOf” and “allOf”

Telemetry Aware Scheduling - *Workflow*



Telemetry Aware Scheduling - *Policy Strategies*

```
apiVersion: telemetry.intel.com/v1alpha1
kind: TASPolicy
metadata:
  name: testing-policy
  namespace: default
spec:
  strategies:
    dontschedule:
      rules:
      - metricname: health_metric
        operator: Equals
        target: 1
    scheduleonmetric:
      rules:
      - metricname: temperature
        operator: LessThan
    labeling:
      rules:
      - metricname: memory_used_card0
        operator: GreaterThan
        target: 100
        labels: ["card0=true"]
      - metricname: memory_used_card1
        operator: GreaterThan
        target: 200
        labels: ["card1=true"]
    deschedule:
      logicalOperator: allOf
      rules:
      - metricName: temperature
        operator: GreaterThan
        target: 80
      - metricName: freeRAM
        operator: LessThan
        target: 200
```

dontschedule: A pod with this strategy will not be scheduled on a node breaking these metric rules.

scheduleonmetric: A single rule strategy that prioritizes nodes based on a comparator and an up-to-date metric value.

labeling: Nodes are labelled based on rule violations. The labels are customizable and can be used to support external components.

deschedule: If a pod with this policy is running on a node that violates it can be de-scheduled with the k8s Descheduler.

Telemetry Aware Scheduling - *Policy Strategies*

```
apiVersion: telemetry.intel.com/v1alpha1
kind: TASPolicy
metadata:
  name: testing-policy
  namespace: default
spec:
  strategies:
    dontschedule:
      rules:
      - metricname: health_metric
        operator: Equals
        target: 1
    scheduleonmetric:
      rules:
      - metricname: temperature
        operator: LessThan
    labeling:
      rules:
      - metricname: memory_used_card0
        operator: GreaterThan
        target: 100
        labels: ["card0=true"]
      - metricname: memory_used_card1
        operator: GreaterThan
        target: 200
        labels: ["card1=true"]
    deschedule:
      logicalOperator: allOf
      rules:
      - metricName: temperature
        operator: GreaterThan
        target: 80
      - metricName: freeRAM
        operator: LessThan
        target: 200
```

dontschedule: A pod with this strategy will not be scheduled on a node breaking these metric rules.

scheduleonmetric: A single rule strategy that prioritizes nodes based on a comparator and an up-to-date metric value.

labeling: Nodes are labelled based on rule violations. The labels are customizable and can be used to support external components.

deschedule: If a pod with this policy is running on a node that violates it can be de-scheduled with the k8s Descheduler.

Telemetry Aware Scheduling - *Policy Strategies*

```yaml
apiVersion: telemetry.intel.com/v1alpha1
kind: TASPolicy
metadata:
  name: testing-policy
  namespace: default
spec:
  strategies:
    dontschedule:
      rules:
      - metricname: health_metric
        operator: Equals
        target: 1
    scheduleonmetric:
      rules:
      - metricname: temperature
        operator: LessThan
    labeling:
      rules:
      - metricname: memory_used_card0
        operator: GreaterThan
        target: 100
        labels: ["card0=true"]
      - metricname: memory_used_card1
        operator: GreaterThan
        target: 200
        labels: ["card1=true"]
    deschedule:
      logicalOperator: allOf
      rules:
      - metricName: temperature
        operator: GreaterThan
        target: 80
      - metricName: freeRAM
        operator: LessThan
        target: 200
```

dontschedule: A pod with this strategy will not be scheduled on a node breaking these metric rules.

scheduleonmetric: A single rule strategy that prioritizes nodes based on a comparator and an up-to-date metric value.

labeling: Nodes are labelled based on rule violations. The labels are customizable and can be used to support external components.

deschedule: If a pod with this policy is running on a node that violates it can be de-scheduled with the k8s Descheduler.

Telemetry Aware Scheduling - *Policy Strategies*

```
apiVersion: telemetry.intel.com/v1alpha1
kind: TASPolicy
metadata:
  name: testing-policy
  namespace: default
spec:
  strategies:
    dontschedule:
      rules:
      - metricname: health_metric
        operator: Equals
        target: 1
    scheduleonmetric:
      rules:
      - metricname: temperature
        operator: LessThan
    labeling:
      rules:
      - metricname: memory_used_card0
        operator: GreaterThan
        target: 100
        labels: ["card0=true"]
      - metricname: memory_used_card1
        operator: GreaterThan
        target: 200
        labels: ["card1=true"]
    deschedule:
      logicalOperator: allOf
      rules:
      - metricName: temperature
        operator: GreaterThan
        target: 80
      - metricName: freeRAM
        operator: LessThan
        target: 200
```

dontschedule: A pod with this strategy will not be scheduled on a node breaking these metric rules.

scheduleonmetric: A single rule strategy that prioritizes nodes based on a comparator and an up-to-date metric value.

labeling: Nodes are labelled based on rule violations. The labels are customizable and can be used to support external components.

deschedule: If a pod with this policy is running on a node that violates it can be de-scheduled with the k8s Descheduler.

Telemetry Aware Scheduling - *Policy Strategies*

```
apiVersion: telemetry.intel.com/v1alpha1
kind: TASPolicy
metadata:
  name: testing-policy
  namespace: default
spec:
  strategies:
    dontschedule:
      rules:
      - metricname: health_metric
        operator: Equals
        target: 1
    scheduleonmetric:
      rules:
      - metricname: temperature
        operator: LessThan
    labeling:
      rules:
      - metricname: memory_used_card0
        operator: GreaterThan
        target: 100
        labels: ["card0=true"]
      - metricname: memory_used_card1
        operator: GreaterThan
        target: 200
        labels: ["card1=true"]
    deschedule:
      logicalOperator: allOf
      rules:
      - metricName: temperature
        operator: GreaterThan
        target: 80
      - metricName: freeRAM
        operator: LessThan
        target: 200
```

dontschedule: A pod with this strategy will not be scheduled on a node breaking these metric rules.

scheduleonmetric: A single rule strategy that prioritizes nodes based on a comparator and an up-to-date metric value.

labeling: Nodes are labelled based on rule violations. The labels are customizable and can be used to support external components.

deschedule: If a pod with this policy is running on a node that violates it can be de-scheduled with the k8s Descheduler.

Demo Time!

Application Deployment File

```
apiVersion: apps/v1
kind: Deployment
metadata:
  name: demo-app-label
  labels:
    app: demo-label
spec:
  replicas: 5
  selector:
    matchLabels:
      app: demo-label
  template:
    metadata:
      labels:
        app: demo-label
        telemetry-policy: labeling-policy
    spec:
      containers:
      - name: nginx
        image: nginx:latest
        imagePullPolicy: IfNotPresent
        resources:
          limits:
            telemetry/scheduling: 1
      affinity:
        nodeAffinity:
          requiredDuringSchedulingIgnoredDuringExecution:
            nodeSelectorTerms:
              - matchExpressions:
                  - key:  telemetry.aware.scheduling.labeling-policy/card0
                    operator: NotIn
                    values:
                      - "true"
                  - key:  telemetry.aware.scheduling.labeling-policy/card1
                    operator: NotIn
                    values:
                      - "true“
```

Telemetry Aware Scheduling Policy

```
apiVersion: telemetry.intel.com/v1alpha1
kind: TASPolicy
metadata:
  name: labeling-policy
  namespace: default
spec:
  strategies:
    dontschedule:
      rules:
      - metricname: health_metric
        operator: Equals
        target: 1
    scheduleonmetric:
      rules:
      - metricname: temperature
        operator: LessThan
    labeling:
      rules:
      - metricname: memory_used_card0
        operator: GreaterThan
        target: 100
        labels: ["card0=true"]
      - metricname: memory_used_card1
        operator: GreaterThan
        target: 200
        labels: ["card1=true"]
```

Descheduler Policy

```
apiVersion: "descheduler/v1alpha1"
kind: "DeschedulerPolicy"
strategies:
   "RemovePodsViolatingNodeAffinity":
        enabled: true
        params:
           nodeAffinityType:
         - "requiredDuringSchedulingIgnoredDuringExecution“
```

Telemetry Aware Scheduling

https://github.com/intel/platform-aware-scheduling/tree/master/telemetry-aware-scheduling

Get Involved:

- Can submit PRs for changes, or enter issues
- Contact us for new feature requests directly via Kubernetes slack or email

Contact:
denisio.togashi@intel.com
madalina.lazar@intel.com
marlow.weston@intel.com

Telemetry Aware Scheduling

https://github.com/intel/platform-aware-scheduling/tree/master/telemetry-aware-scheduling

White papers:

D. Togashi, T. Shah, M. Weston
https://builders.intel.com/docs/networkbuilders/facilitating-consumption-of-kubernetes-extenders-with-telemetry-aware-scheduler-technology-guide-1632427976.pdf

D. Cremins, K. Muldoon, S. Sehgal
https://builders.intel.com/docs/networkbuilders/telemetry-aware-scheduling-automated-workload-optimization-with-kubernetes-k8s-technology-guide.pdf

Q & A