

→ Dynamic Resource Allocation (DRA)
→ Resource based scheduling

Pod (AI workload needing GPU)

```
spec:
  nodeSelector:
    nvidia.com/gpu.present: "true"

  tolerations:
    - key: "nvidia.com/gpu"
      operator: "Exists"
      effect: "NoSchedule"

  containers:
    - name: server
      image: myrepo/llm-server:latest
      resources:
        limits:
          nvidia.com/gpu: 1
```

Pod (AI workload needing GPU)

```
spec:
  affinity:
    nodeAffinity:
      requiredDuringSchedulingIgnoredDuringExecution:
        nodeSelectorTerms:
          - matchExpressions:
              - key: nvidia.com/gpu.memory
                operator: Gt
                values: ["40000"]

      preferredDuringSchedulingIgnoredDuringExecution:
        - weight: 50
          preference:
            matchExpressions:
              - key: nvidia.com/gpu.count
                operator: Gt
                values: ["1"]

  tolerations:
    - key: "nvidia.com/gpu"
      operator: "Exists"
      effect: "NoSchedule"

  containers:
    - name: server
      image: myrepo/llm-server:latest
      resources:
        limits:
          nvidia.com/gpu: 1
```

Pod (AI workload needing GPU)

```
kind: ResourceClaimTemplate
metadata:
  name: gpu-template

spec:
  spec:
    devices:
      requests:
        - name: gpu
          exactly:
            deviceClassName: nvidia-gpu
            allocationMode: ExactCount
            count: 1

spec:
  resourceClaims:
    - name: gpu
      resourceClaimTemplateName: gpu-template
  tolerations:
    - key: "nvidia.com/gpu"
      operator: "Exists"
      effect: "NoSchedule"
```

resourceclaim-controller

watch

create/update

ResourceClaim

read

read

ClusterPolicy

```
spec:
  gfd: 1
  enabled: true
  devicePlugin:
    config:
      name: gpu-config
      default: sharing
  mig:
    strategy: mixed
```

ConfigMap
gpu-config

```
data:
  enable/configure either time
  slicing or MIG
```

GPU Operator

watch

create/update

Pod (Non-AI general workload not needing GPU)

```
no taint tolerations
```

Can't be schedule: Pod does NOT tolerate the `nvidia.com/gpu` taint

see node has GPU resource available

create/update

assign Pod to this node

Can be schedule:

- Pod tolerates the `nvidia.com/gpu` taint
- The node ResourceSlice satisfies ResourceClaim needed by the Pod

Can be schedule:

- Pod tolerates the `nvidia.com/gpu` taint
- The node satisfies the nodeAffinity conditions
- Node has enough GPUs

Can be schedule:

- Pod tolerates the `nvidia.com/gpu` taint
- The node satisfies the nodeSelector
- Node has enough GPUs

update
status.capacity
status.allocatable

discover GPU properties and add `nvidia.com/*` labels

scrape GPU metrics

Prometheus

Node

```
metadata:
  name: gpu-node-1
  labels:
    nvidia.com/gpu.present: "true"
    nvidia.com/gpu.memory: "49152"

spec:
  taints:
    - key: nvidia.com/gpu
      value: "true"
      effect: NoSchedule
```

add `nvidia.com/gpu` to `repel` non-AI workloads that do not need GPU from this node

