
KubeCon
CloudNativeCon
Europe 2019
Sig-Autoscaling Deep Dive
Aleksandra Malinowska, Google
Sławomir Chyłek, Google


KubeCon
CloudNativeCon
Europe 2019
Vertical Pod Autoscaler

Vertical Pod Autoscaler

Do you set pod resource request?

Vertical Pod Autoscaler

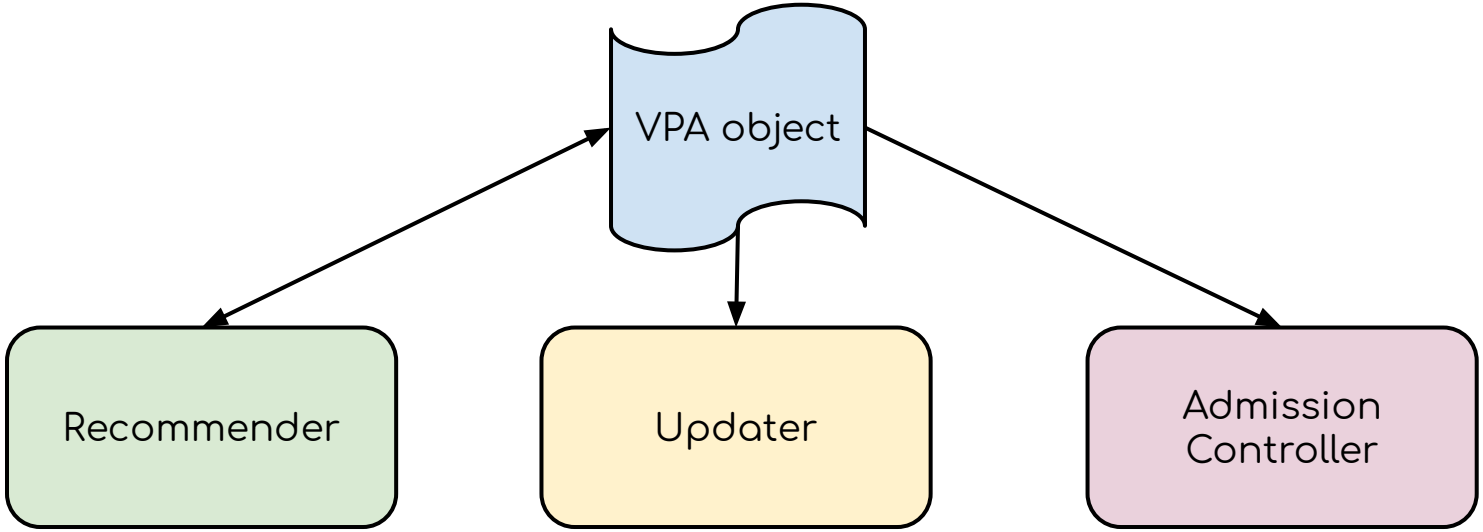
Are your pod request values correct?

Vertical Pod Autoscaler

How Vertical Pod Autoscaler helps you?

- Hands free resource adjustments
- Save money
- Buy reliability
- Increase cluster utilization

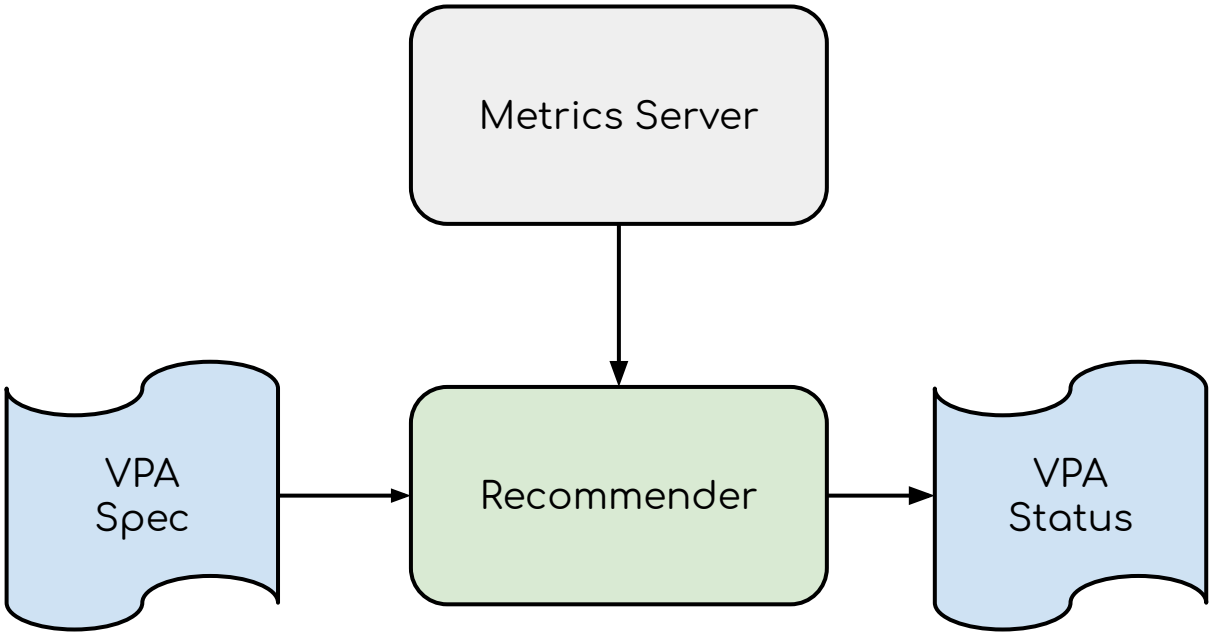
VPA — Components



VPA API

- VPA target
  - Everything with Scale sub-resource
- Modes of actuation
  - Off
  - Initial
  - Auto / Recreate
- Per container/resource configuration
  - On/Off
  - Min/Max

VPA — Recommendations



VPA — Recommendations


```
Status:
  Recommendation:
    Container Recommendations:
      Container Name:  app
      Lower Bound:
        Cpu:     381m
        Memory:  262144k
      Target:
        Cpu:     587m
        Memory:  262144k
      Uncapped Target:
        Cpu:     587m
        Memory:  262144k
      Upper Bound:
        Cpu:     141467m
        Memory:  2771500k
```

VPA — Recommendations

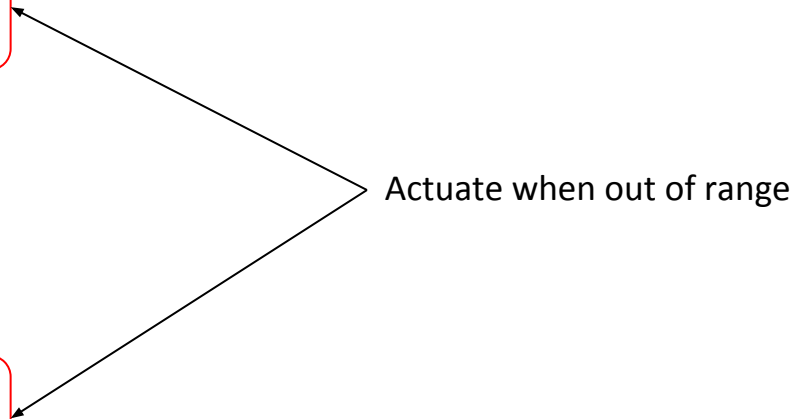

```
Status:
  Recommendation:
    Container Recommendations:
      Container Name:  app
      Lower Bound:
        Cpu:     381m
        Memory:  262144k
      Target:
        Cpu:     587m
        Memory:  262144k
      Uncapped Target:
        Cpu:     587m
        Memory:  262144k
      Upper Bound:
        Cpu:     141467m
        Memory:  2771500k
```

VPA — Recommendations


```
Status:
  Recommendation:
    Container Recommendations:
      Container Name:  app
      Lower Bound:
        Cpu:     381m
        Memory:  262144k
      Target:
        Cpu:     587m
        Memory:  262144k
      Uncapped Target:
        Cpu:     587m
        Memory:  262144k
      Upper Bound:
        Cpu:     141467m
        Memory:  2771500k
```

VPA — Recommendations


```
Status:
  Recommendation:
    Container Recommendations:
      Container Name:  app
      Lower Bound:
        Cpu:      381m
        Memory:   262144k
      Target:
        Cpu:      587m
        Memory:   262144k
      Uncapped Target:
        Cpu:      587m
        Memory:   262144k
      Upper Bound:
        Cpu:      141467m
        Memory:   2771500k
```



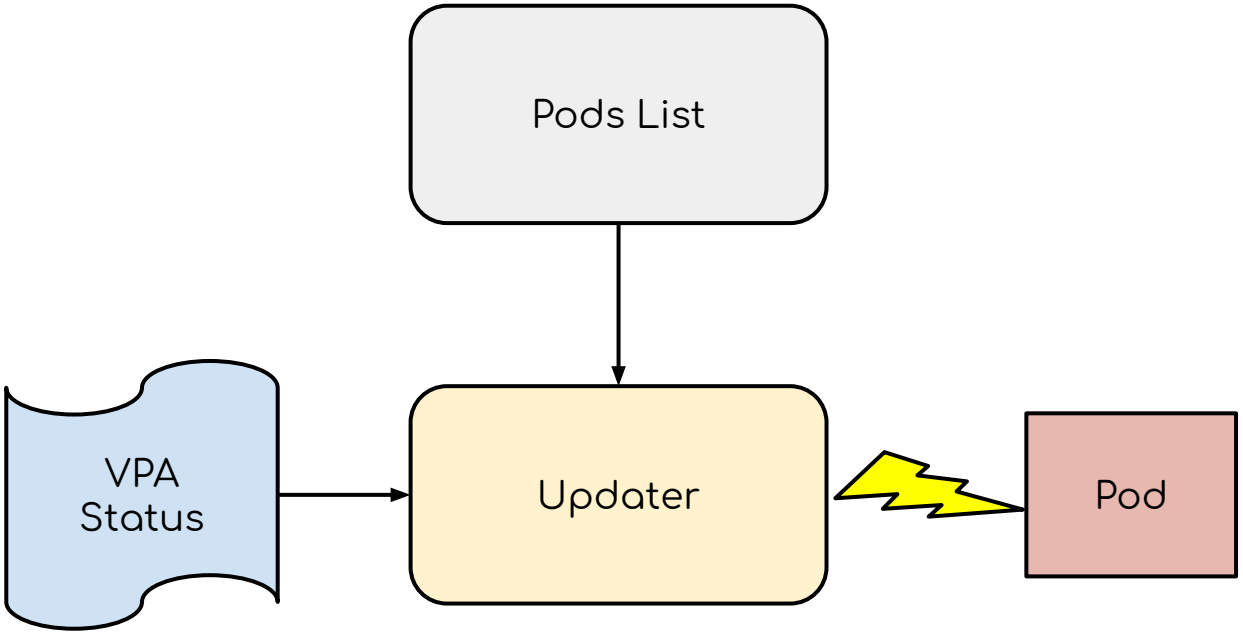
VPA — Recommendations

- Computing recommendations — as of today:
  - Cover 8 days of history
  - Recommendations are silently computed for every possible target
    - Creating VPA object is just surfacing the recommendation
  - Surfaced recommendations are checkpointed
    - To gain restart stability

VPA — Recommendations

- Computing recommendations — as of today:
  - Decaying histogram of weighted samples
    - Newer samples have higher weight (decaying)
    - Recommendation:
      - CPU ~ 90-th percentile
      - Memory ~ max over window
    - Lower/Upper bound
      - Different percentiles (50-th, 95-th)
      - Confidence factor (more samples -> closer to target)
    - Safety margins
  - OOMs - artificial samples with a multiplier

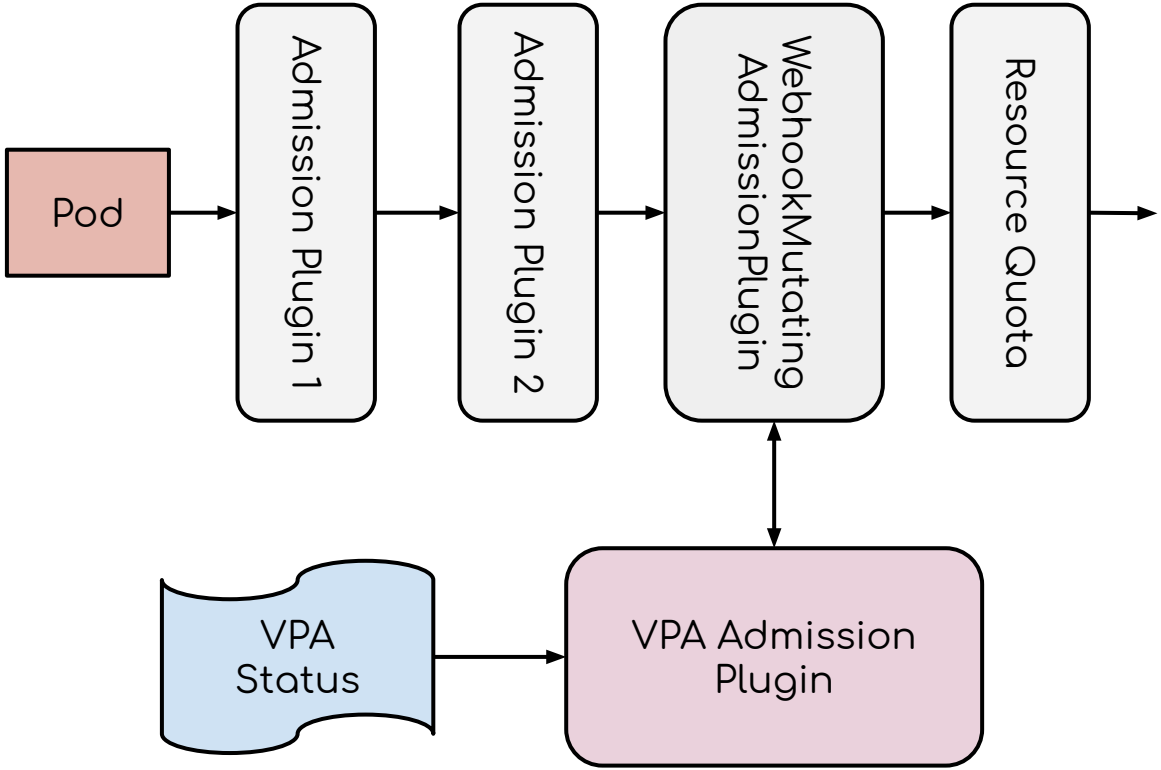
VPA — Updater



VPA — Updater

- Operates only in "Auto" modes
- Eviction is needed to change resource request*
- Uses eviction API -> Pod Disruption Budget (PDB) is respected
- Additional restrictions:
  - Min number of replicas (default 2)
  - Eviction tolerance (default: 50% of replicas)
- Pod eviction priority
  - Recent OOMs
  - Pods most offending requests

VPA — Admission Controller



VPA — Admission Controller

- Operates in "Auto"/"Init" modes
- Applies Target Recommendation to requests
- Annotates Pod ("vpaUpdates")

VPA — Best practices for Auto mode

- Use when pods can be restarted
- Start with "Off" mode enabled for 1 week to gain confidence
- Define PDB
- Define min/max resources
- Copy recommendations to pod spec from time to time
- VPA adopts slowly to new usage characteristics
    - e.g. numbers of replicas changed
- Keep metrics server healthy
- Enable Cluster Autoscaler
- Mixing with HPA only when you know what you are doing
    - e.g. HPA based on QPS or absolute value of CPU usage

VPA — Status

- API in Beta2
- Used in production clusters
- Feedback is welcome
- Next steps:
  - Limits
  - GA VPA
  - In-place update


KubeCon
CloudNativeCon
Europe 2019
Cluster Autoscaler

Cluster Autoscaler

Is your cluster large enough to fit all workloads?

Cluster Autoscaler

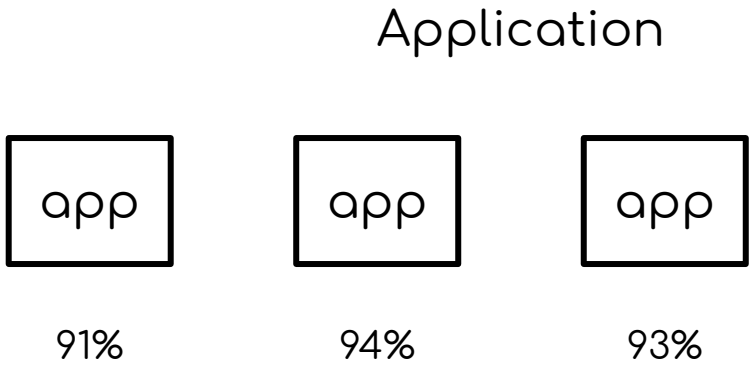
Are your nodes underutilized?

NOT metric based

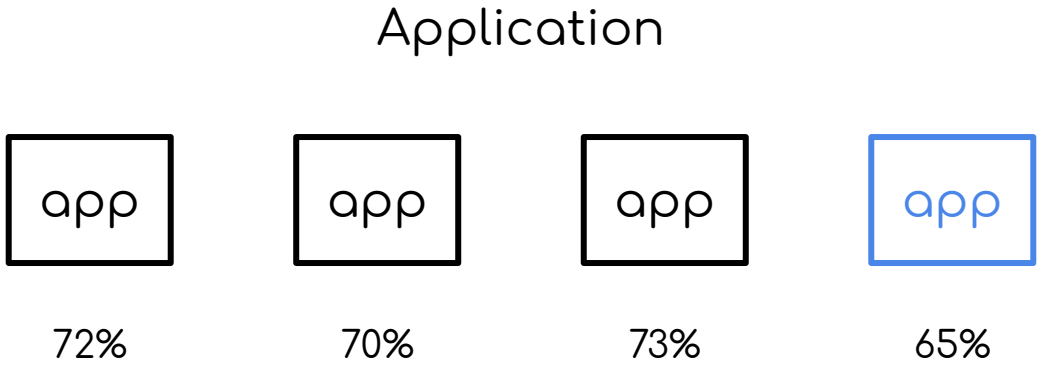
Naive solution:

calculate desired number of nodes based on utilization

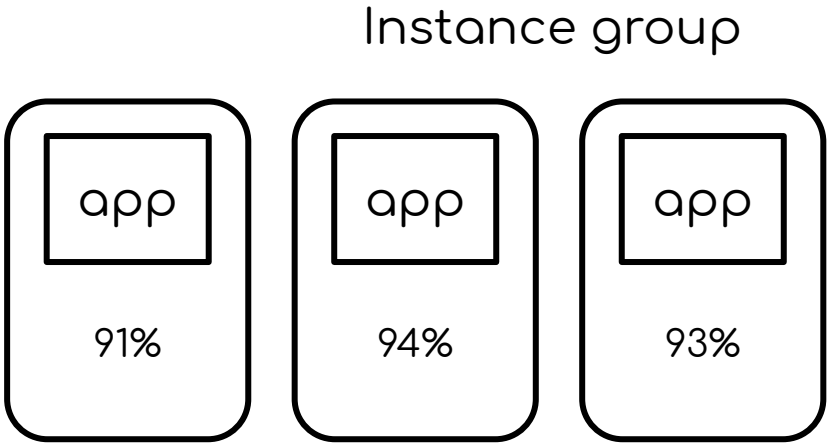
Horizontal scaling



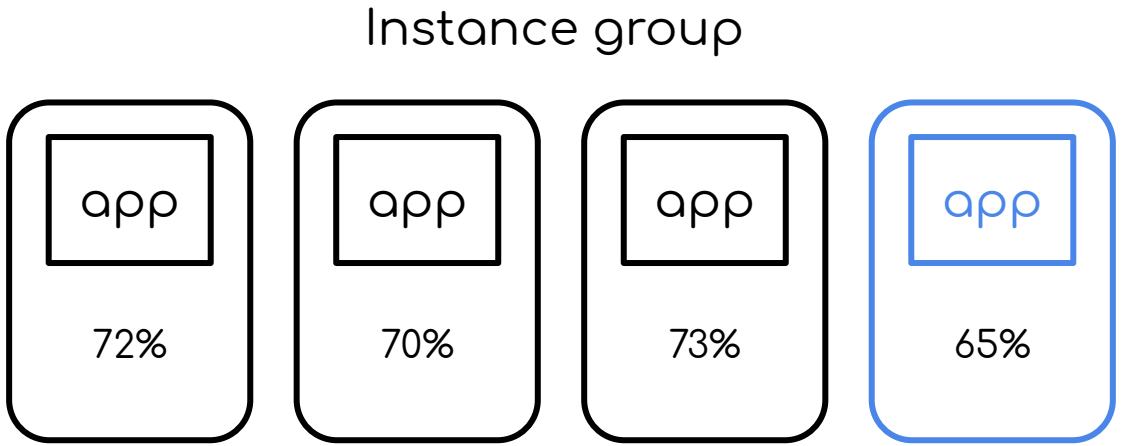
Horizontal scaling



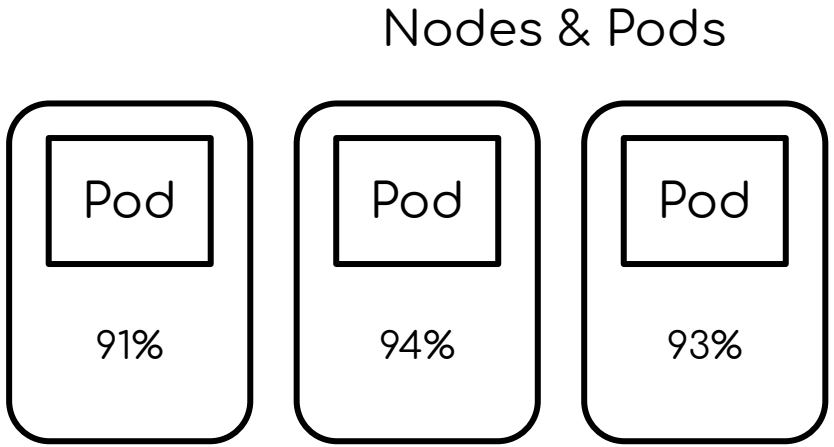
Horizontal scaling



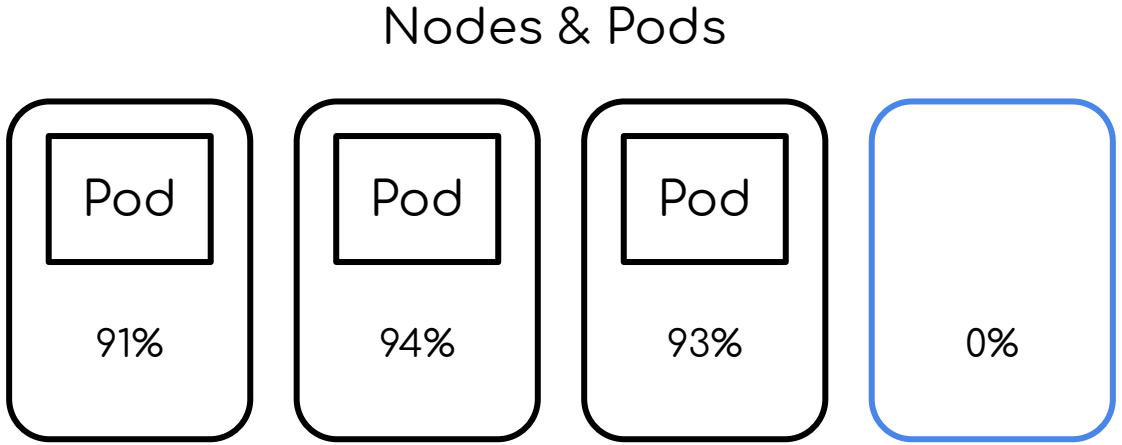
Horizontal scaling



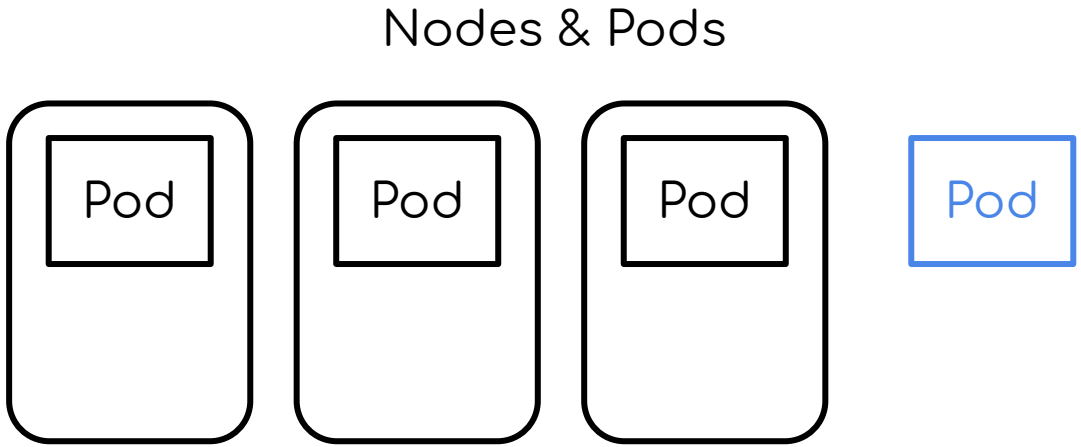
Horizontal scaling



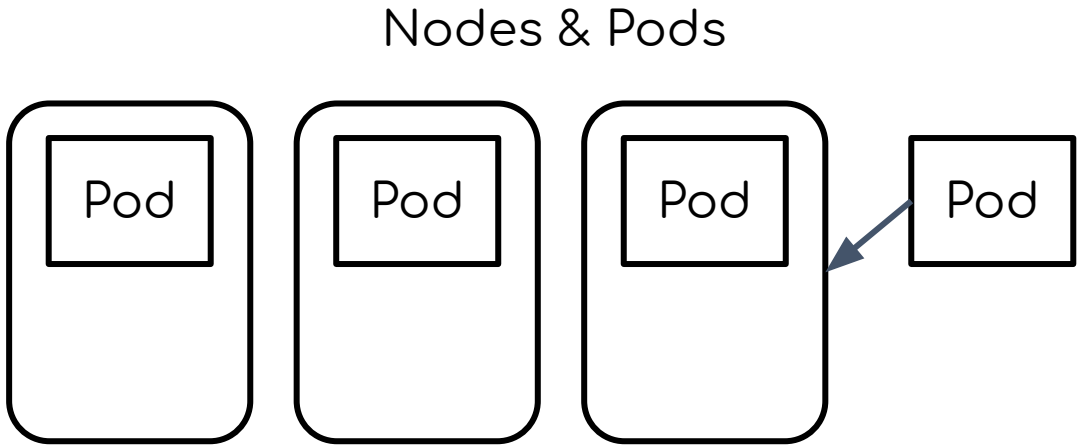
Node scaling?



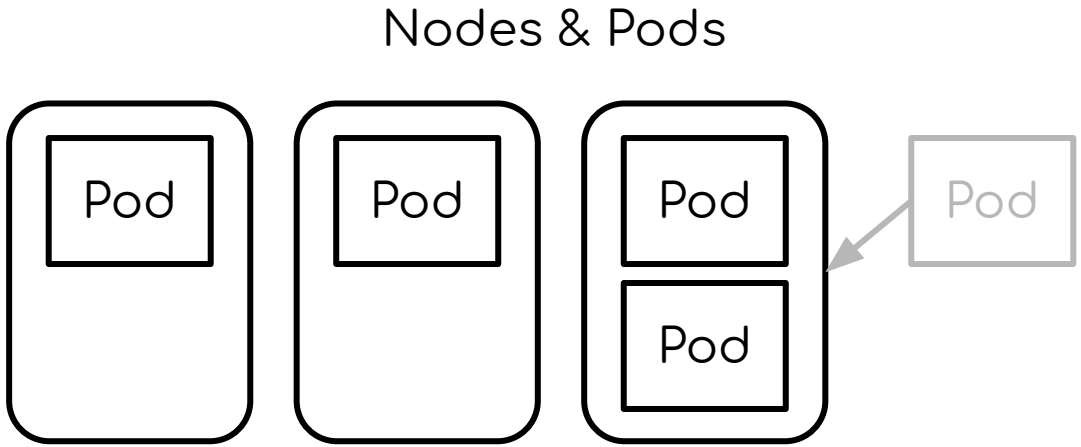
Pod scaling



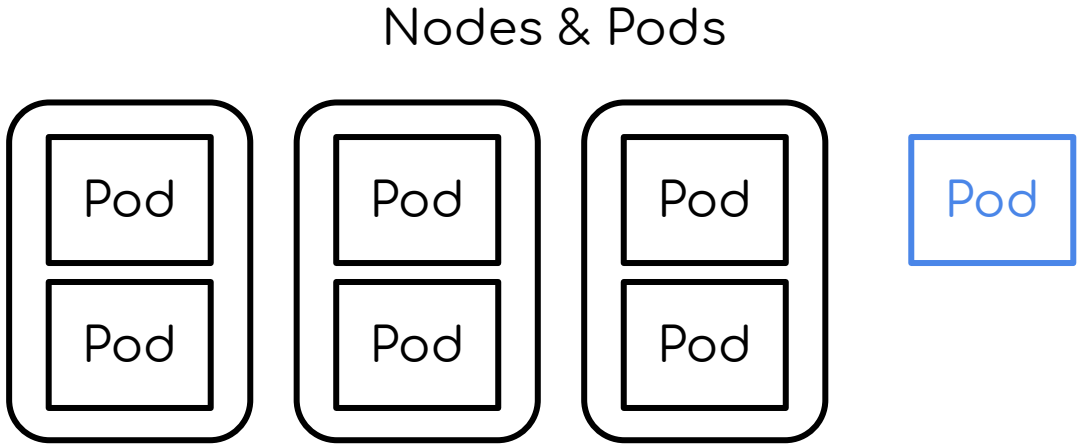
Pod scaling



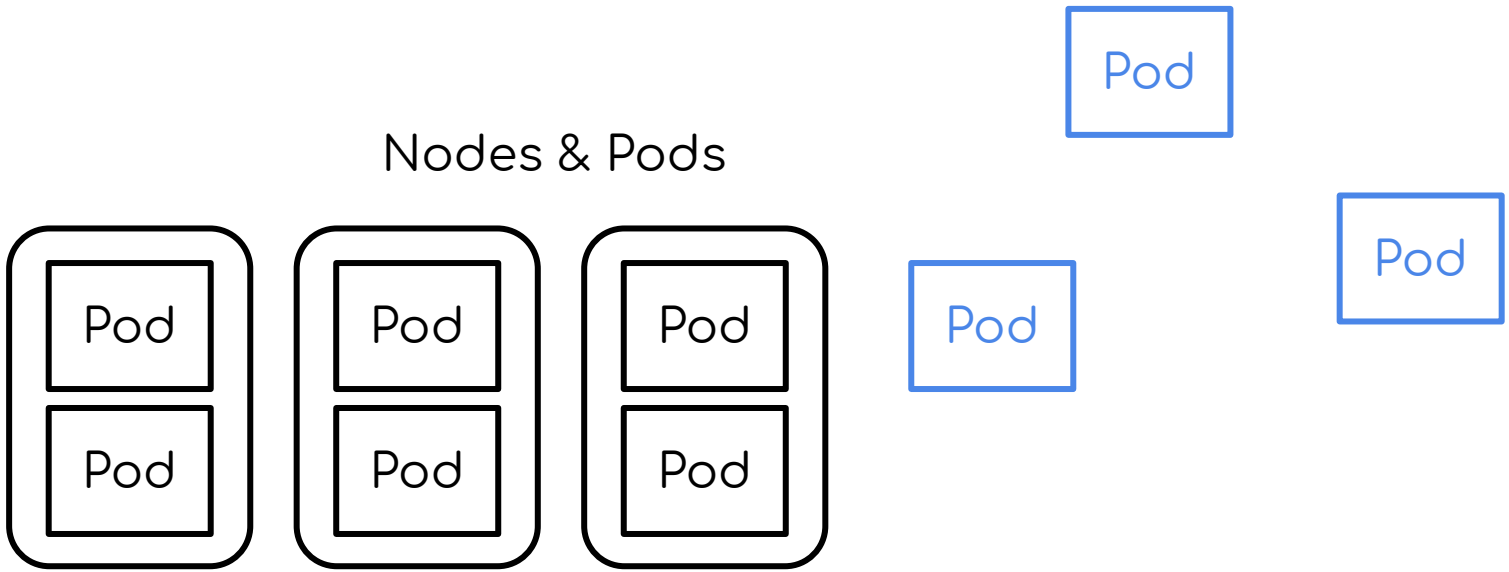
Pod scaling



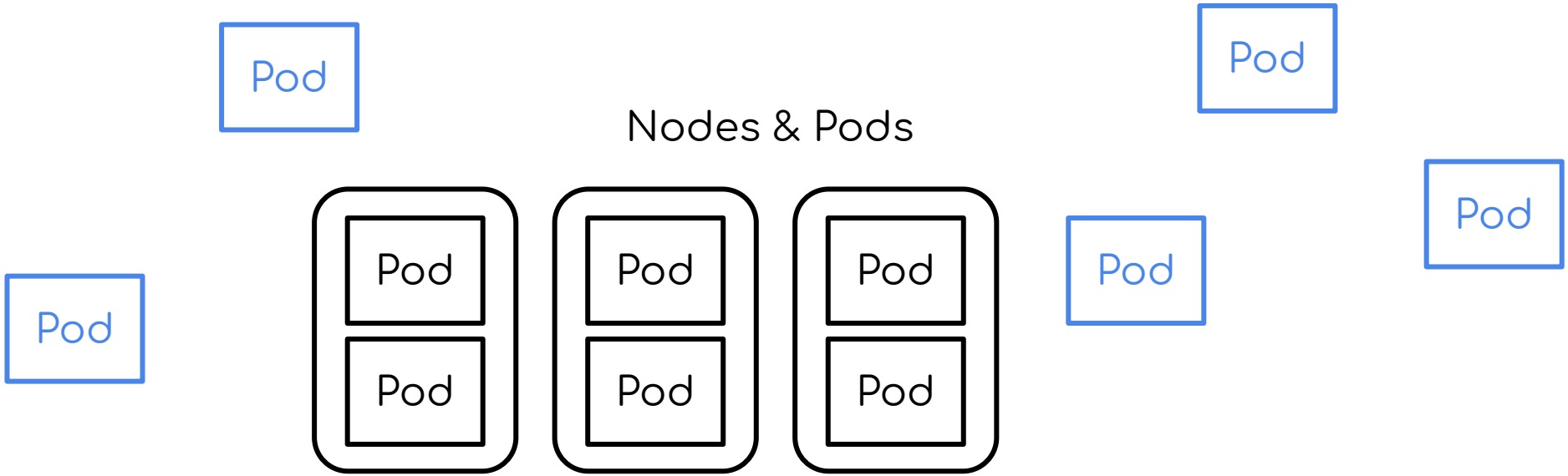
Pending pods



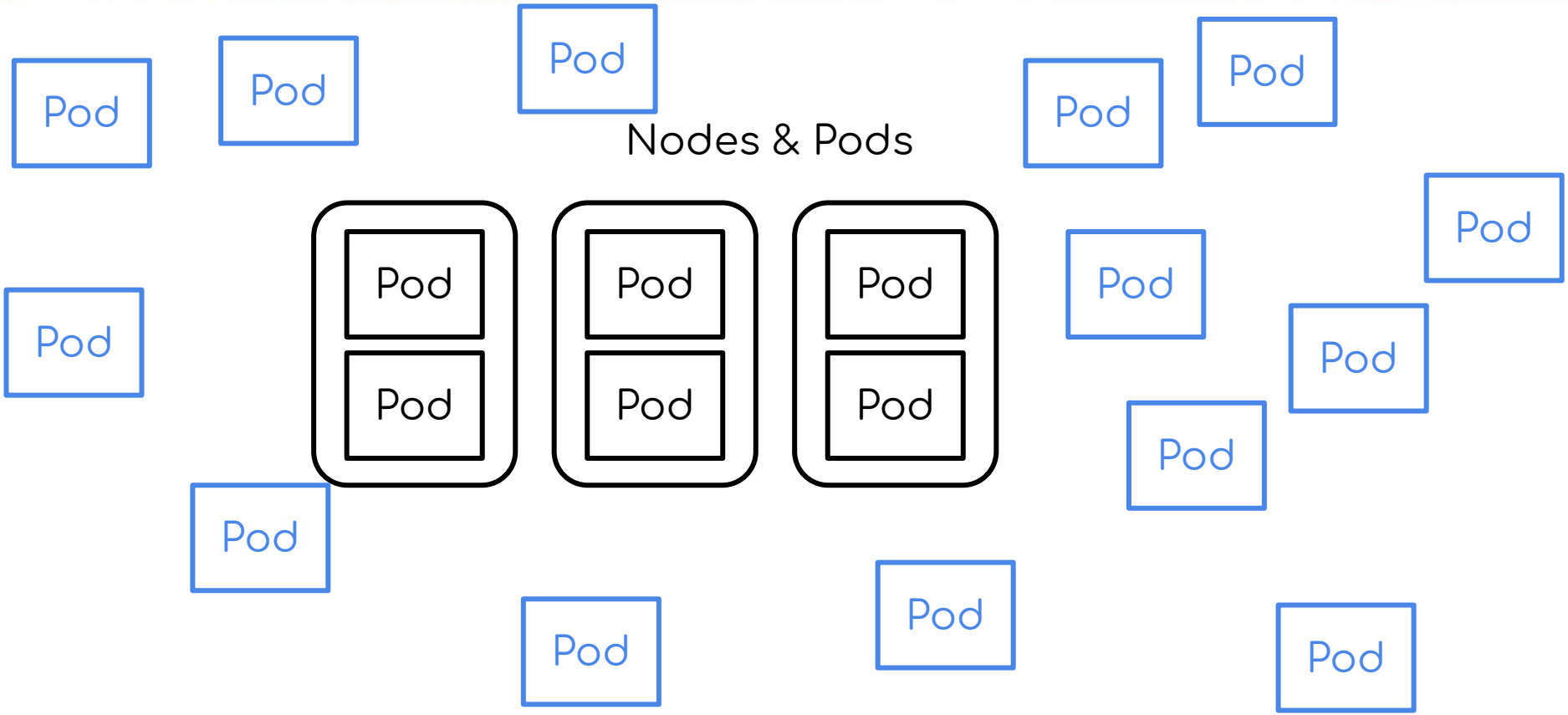
Pending pods



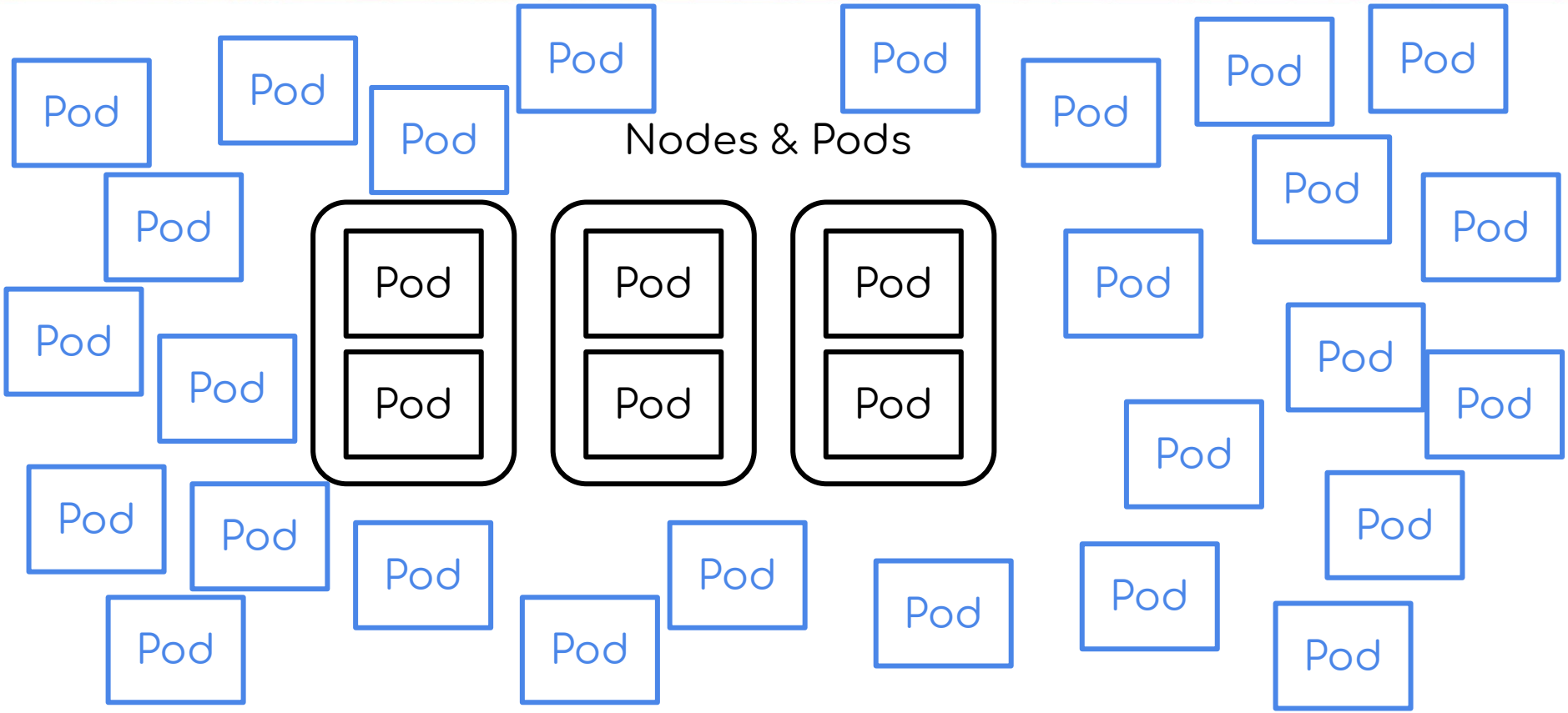
Pending pods



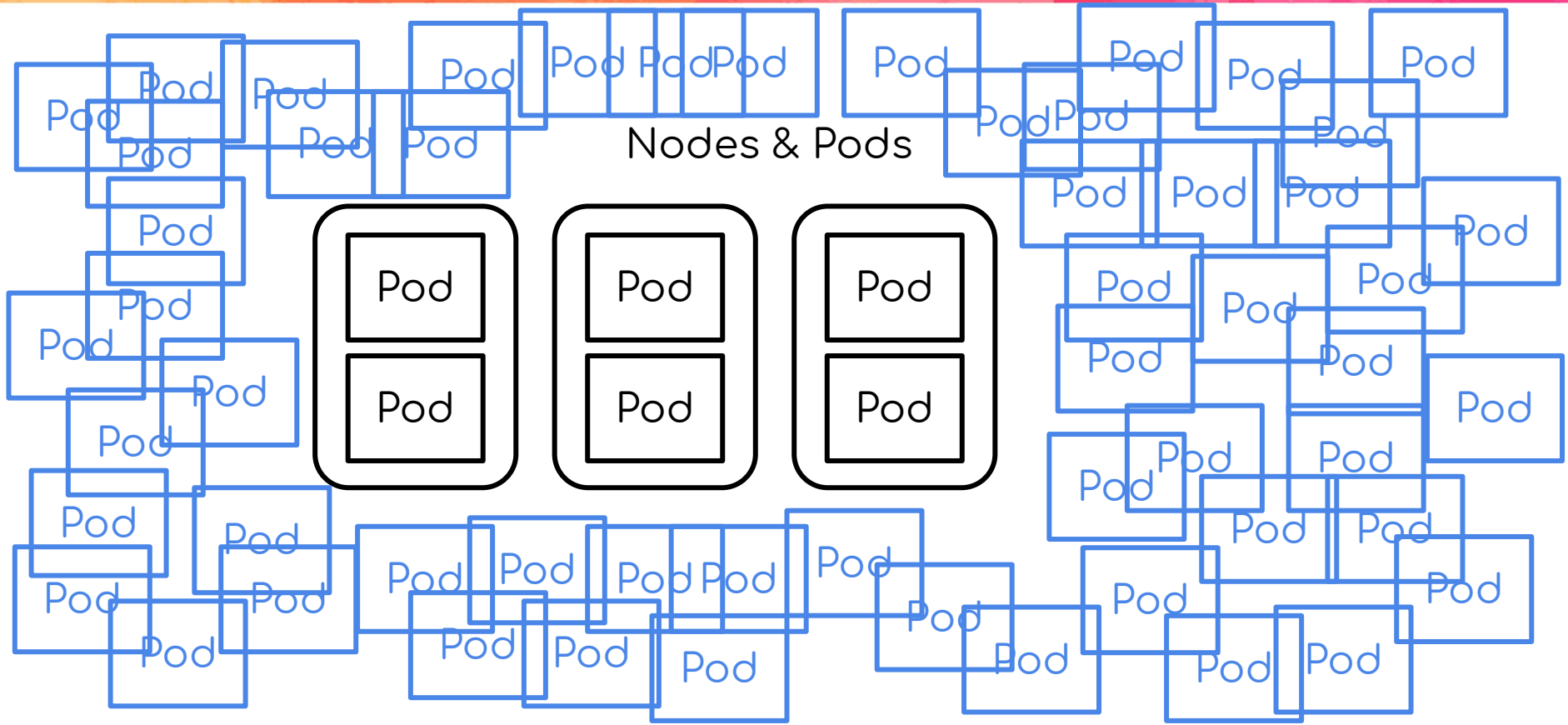
Pending pods



Pending pods



Pending pods



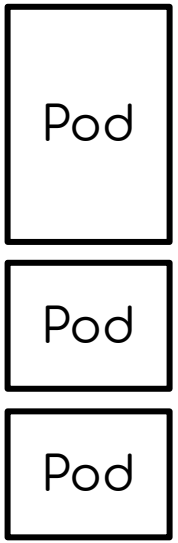
Pending pods

Solution:

add just enough nodes to make pods run

Bin packing

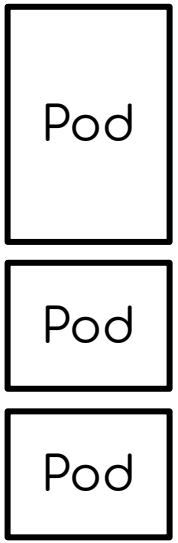
Pending Pods



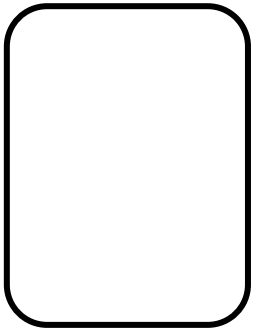
New nodes

Bin packing

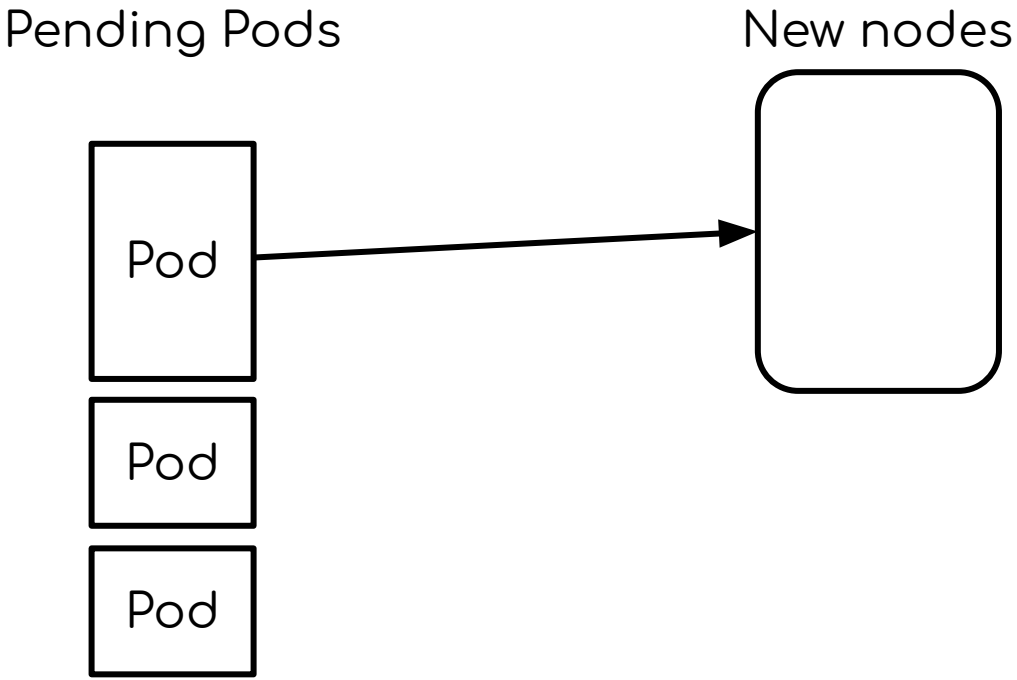
Pending Pods



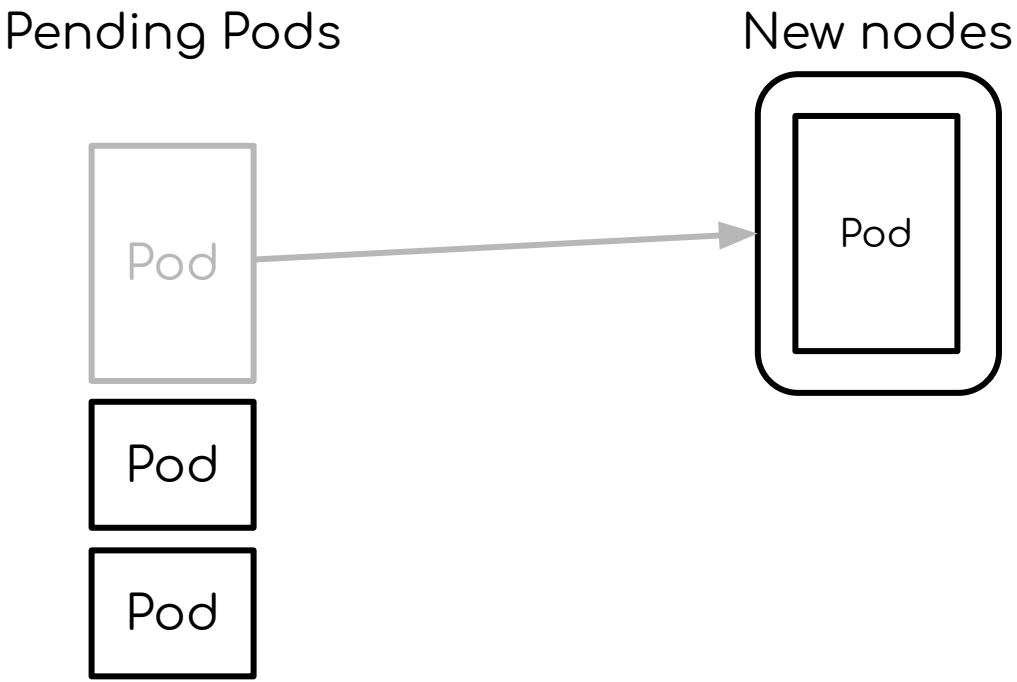
New nodes



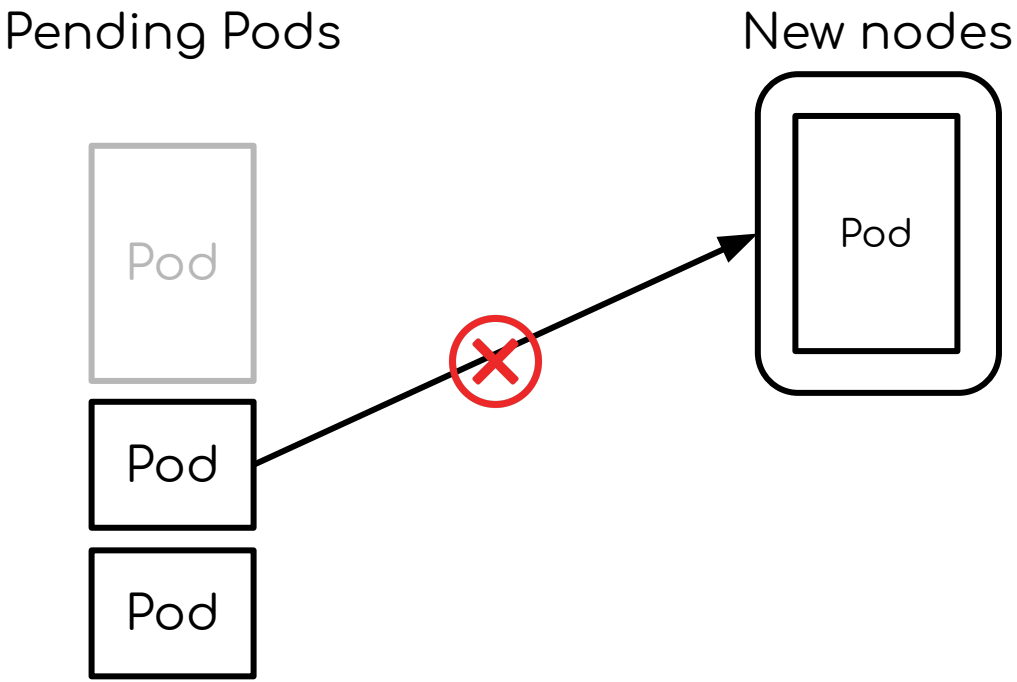
Bin packing



Bin packing

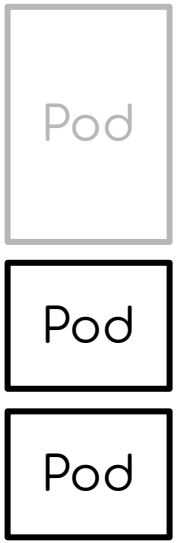


Bin packing

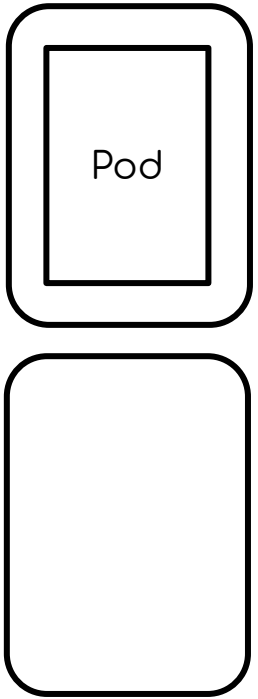


Bin packing

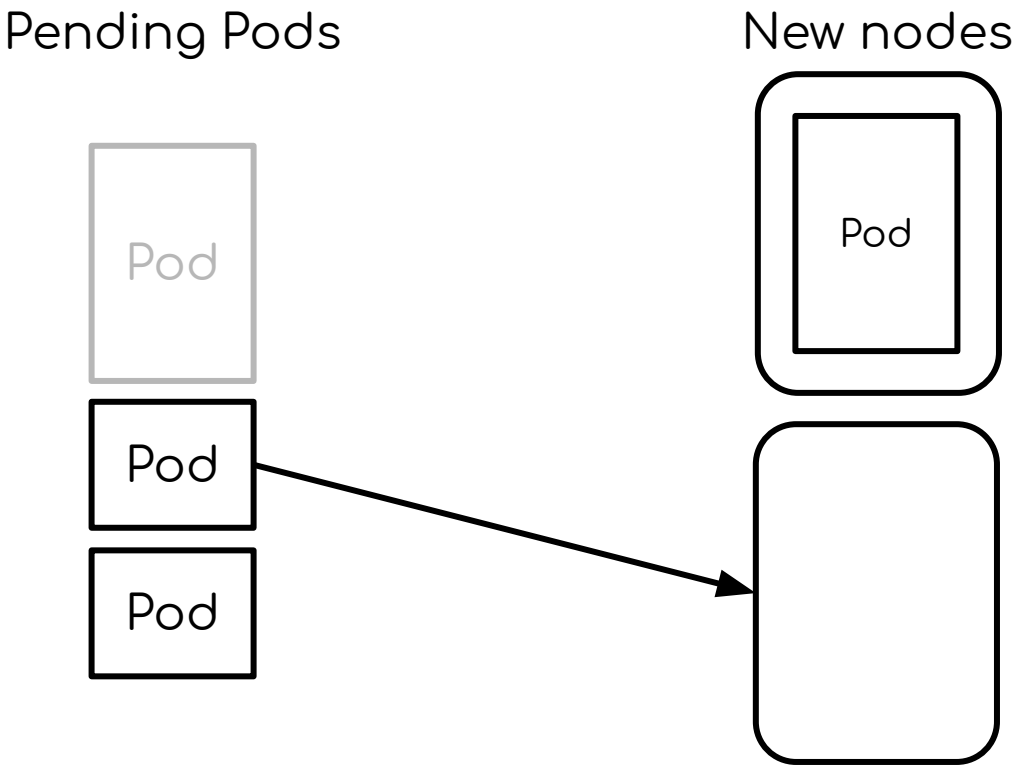
Pending Pods



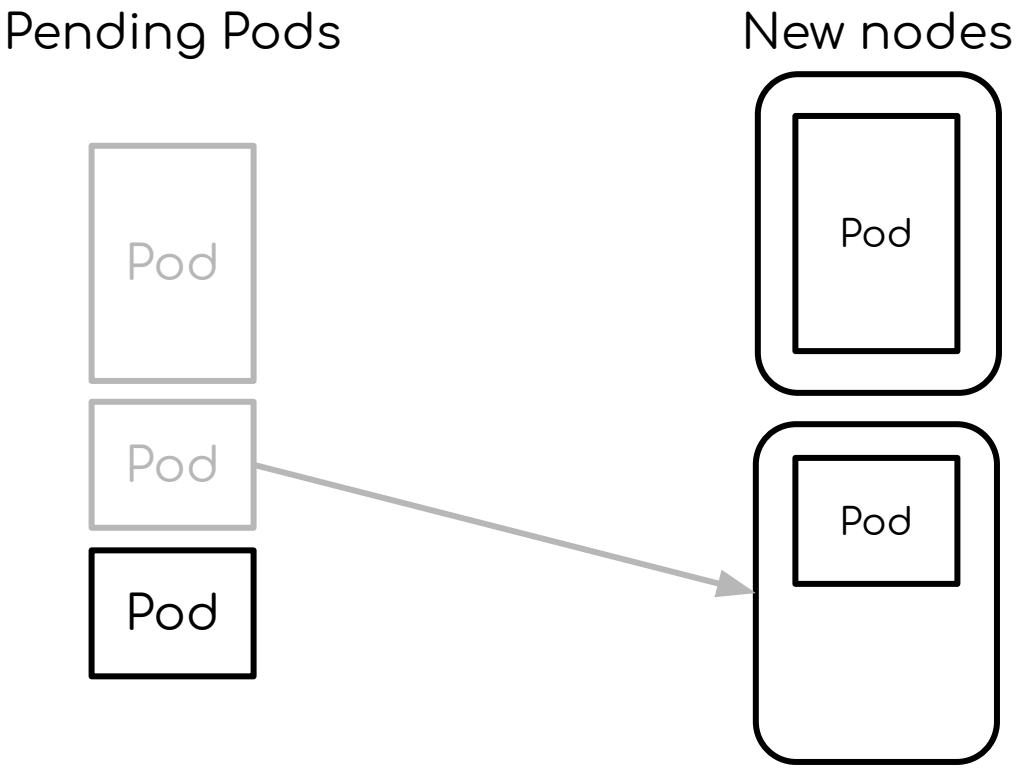
New nodes



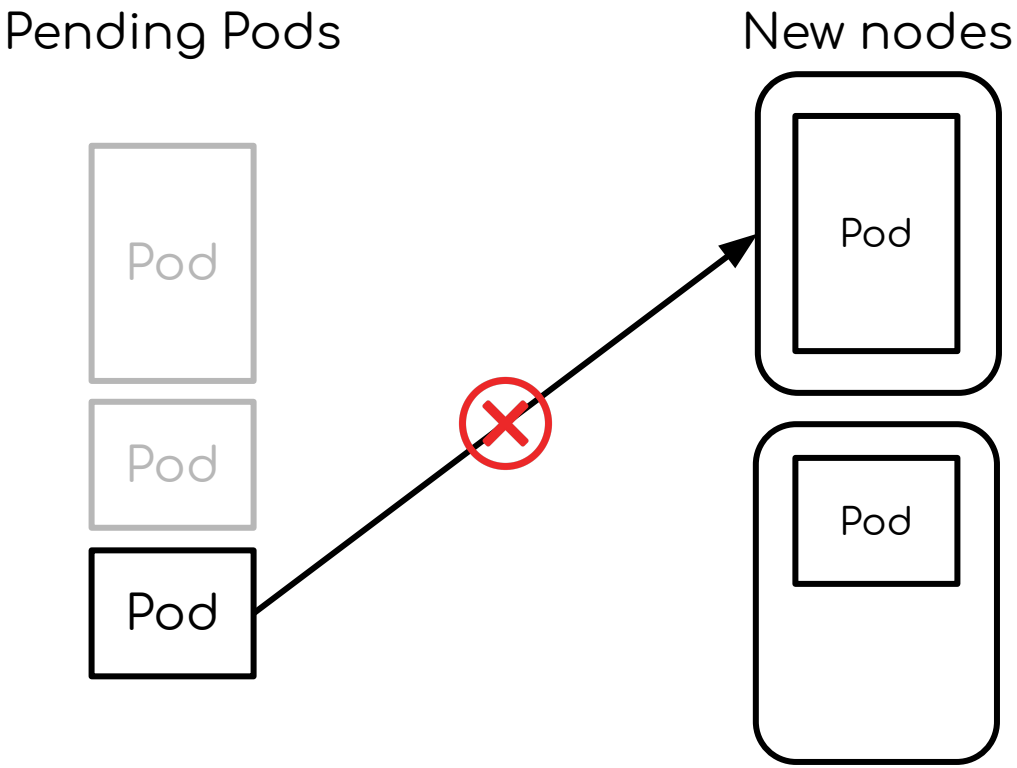
Bin packing



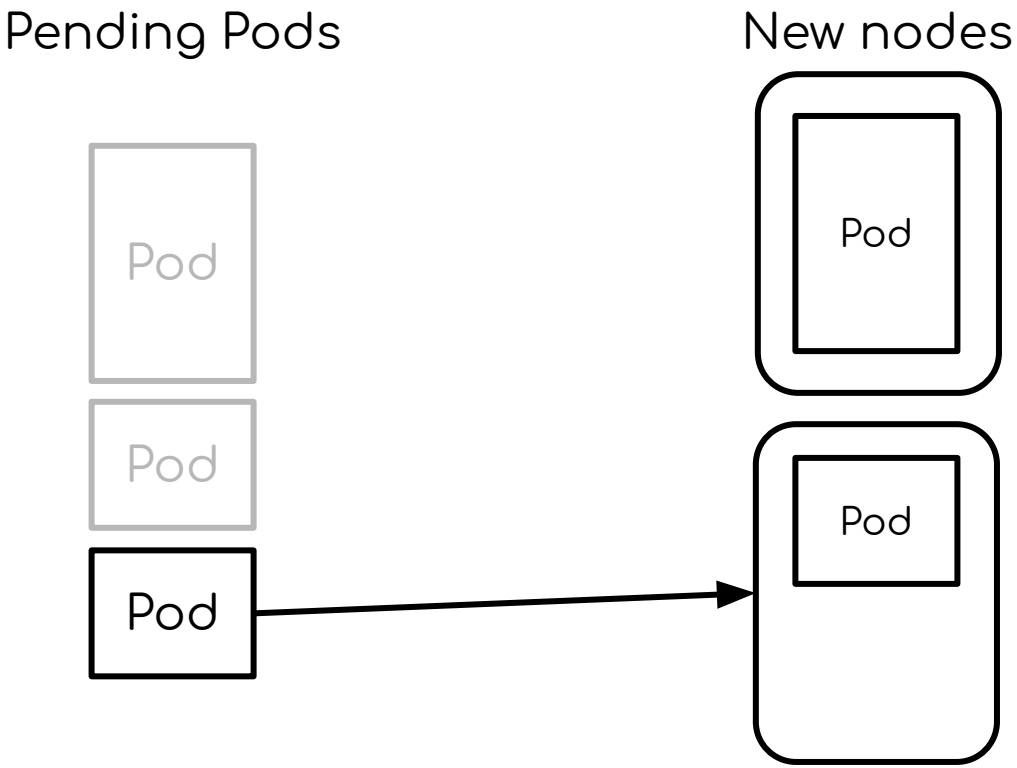
Bin packing



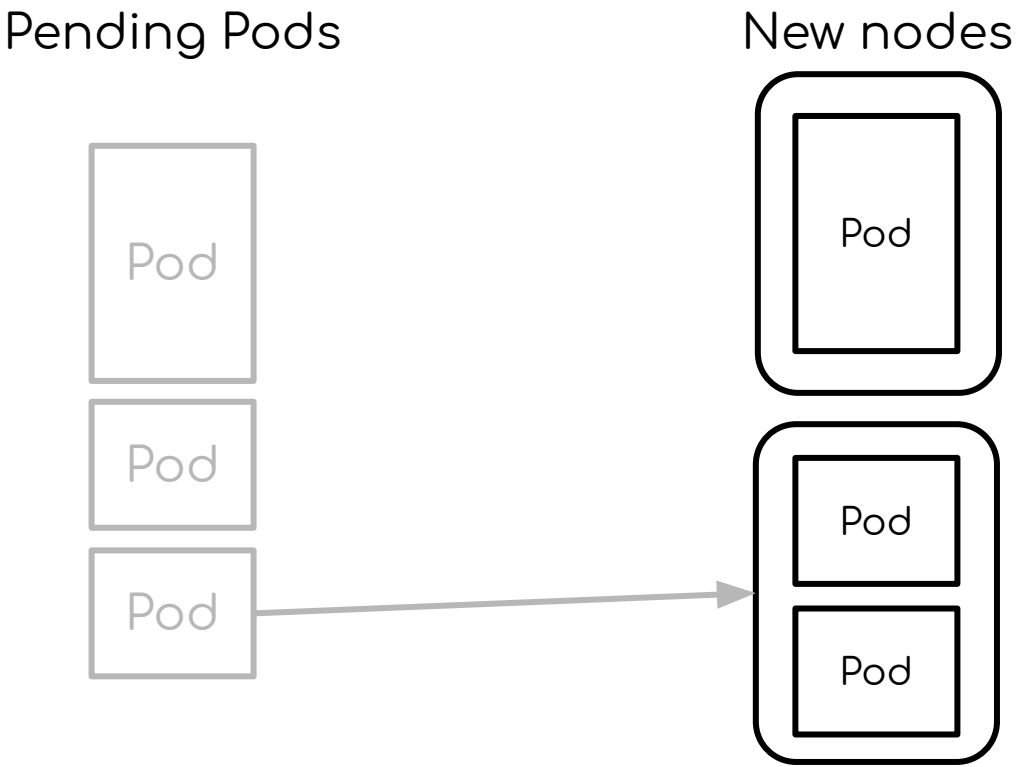
Bin packing



Bin packing



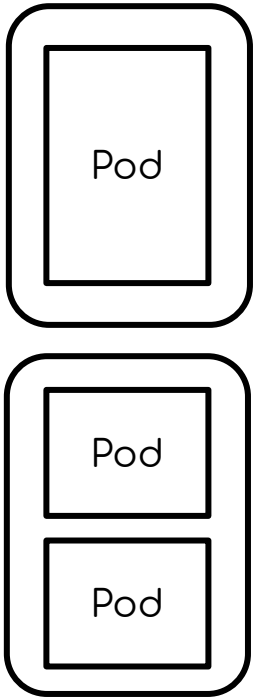
Bin packing



Bin packing

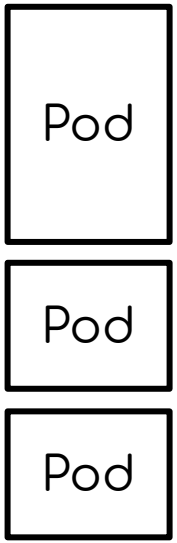
Pending Pods

New nodes



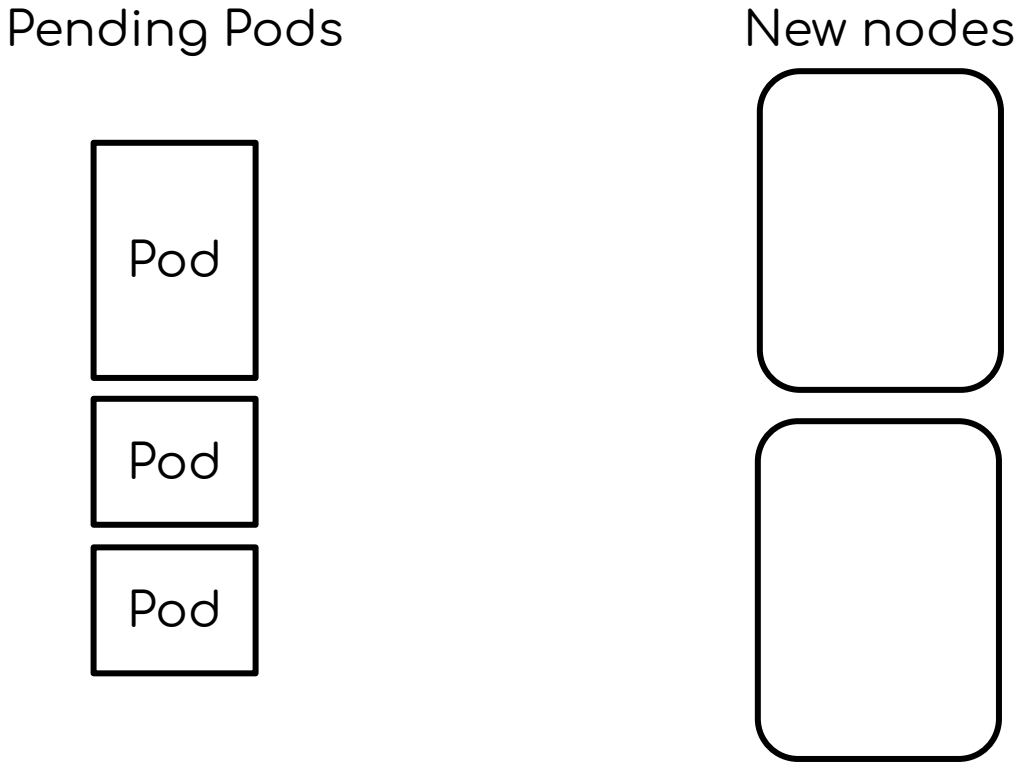
Bin packing

Pending Pods



New nodes

Decision: Add 2 nodes.



Unneeded nodes

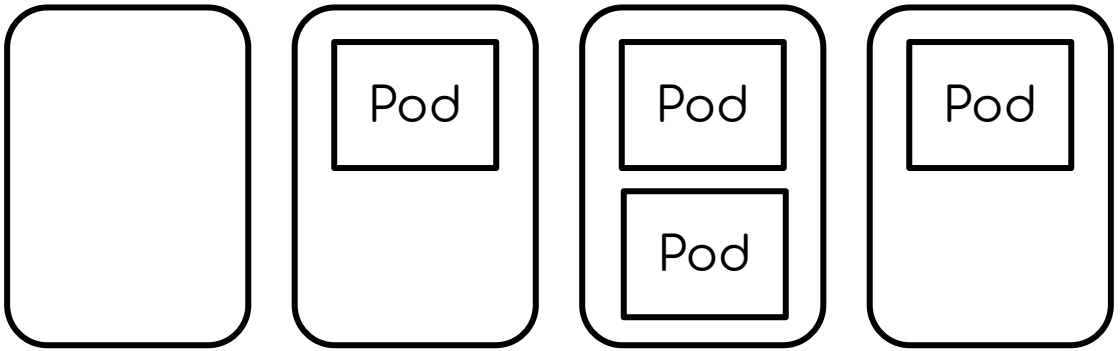
Solution:

add just enough nodes to make pods run

remove nodes only if the pods can still run

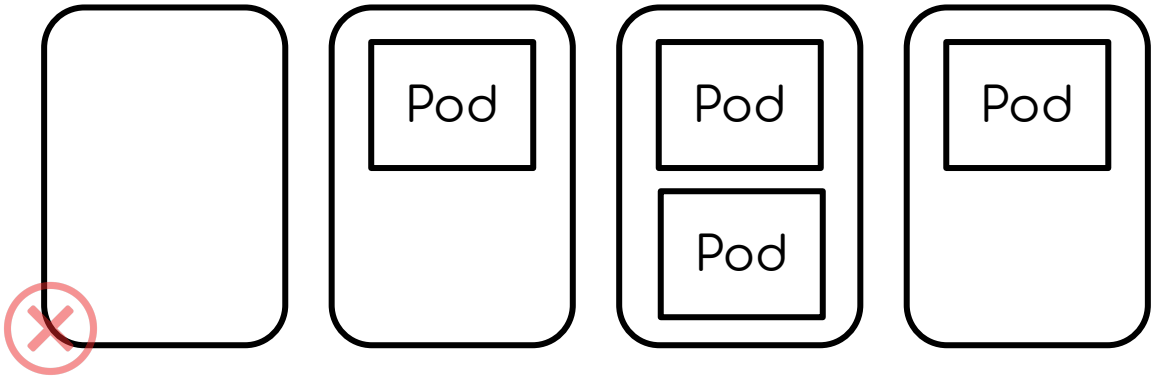
Unneeded nodes

Nodes & Pods



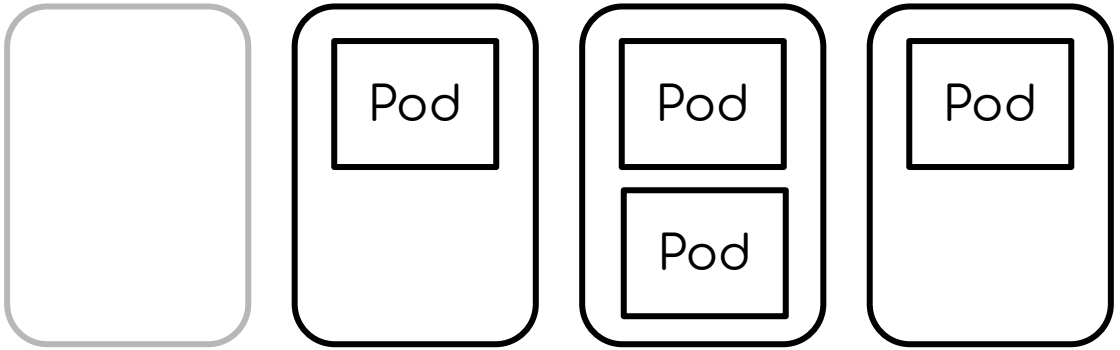
Unneeded nodes

Nodes & Pods

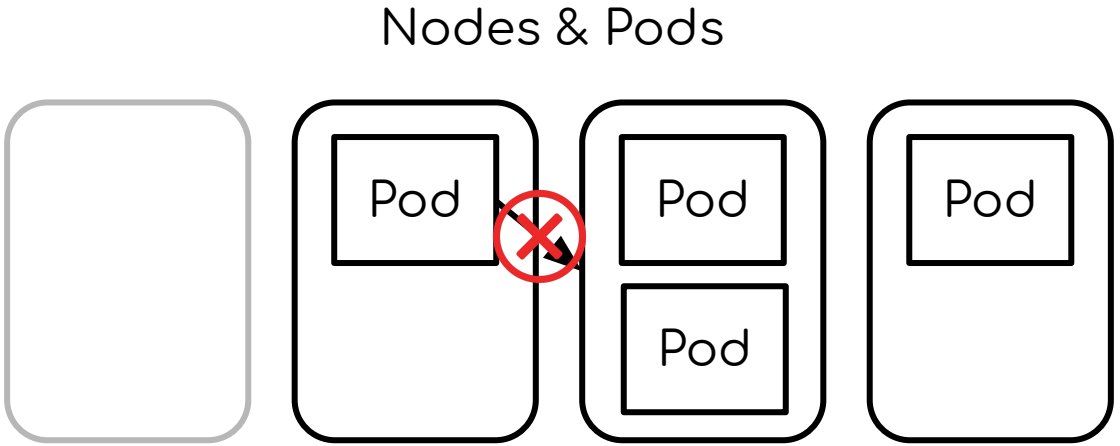


Unneeded nodes

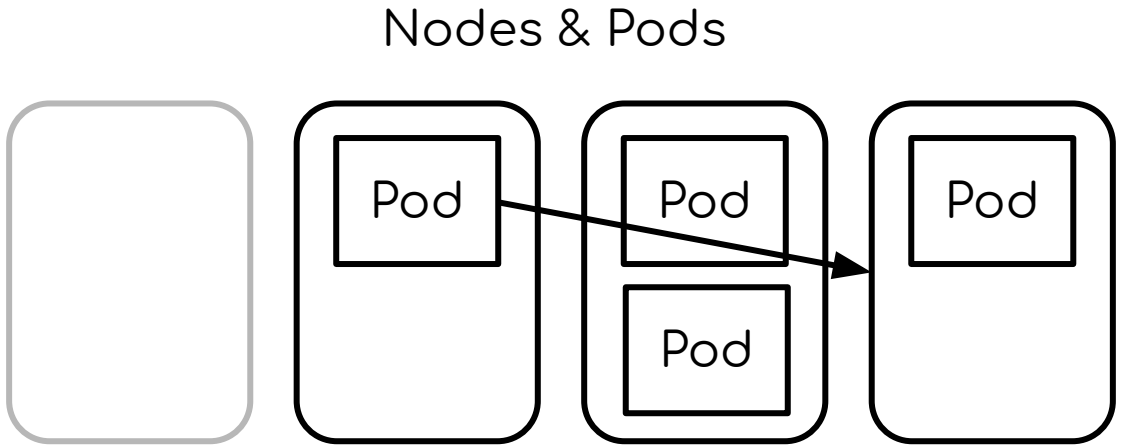
Nodes & Pods



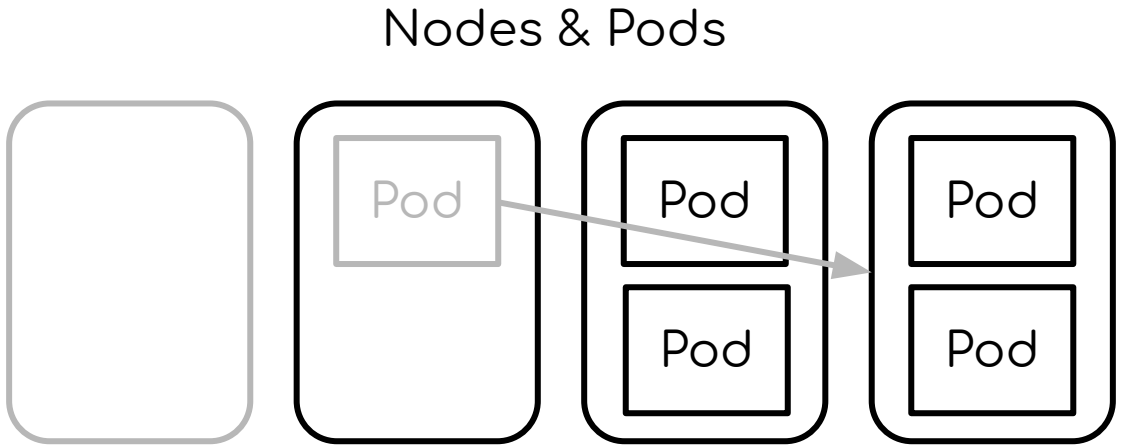
Unneeded nodes



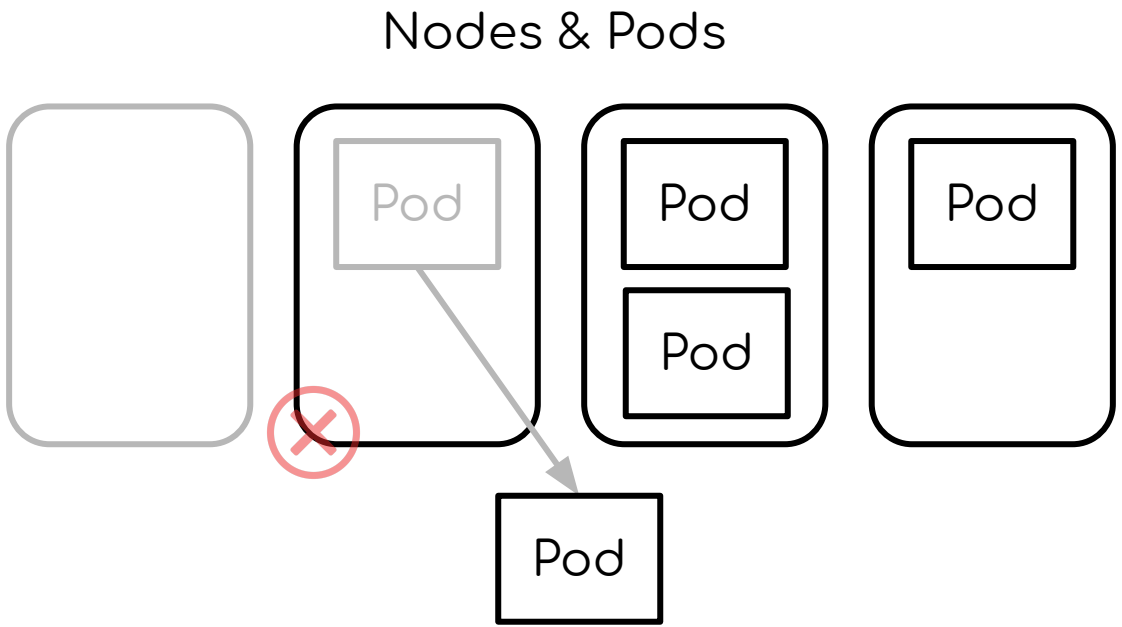
Unneeded nodes



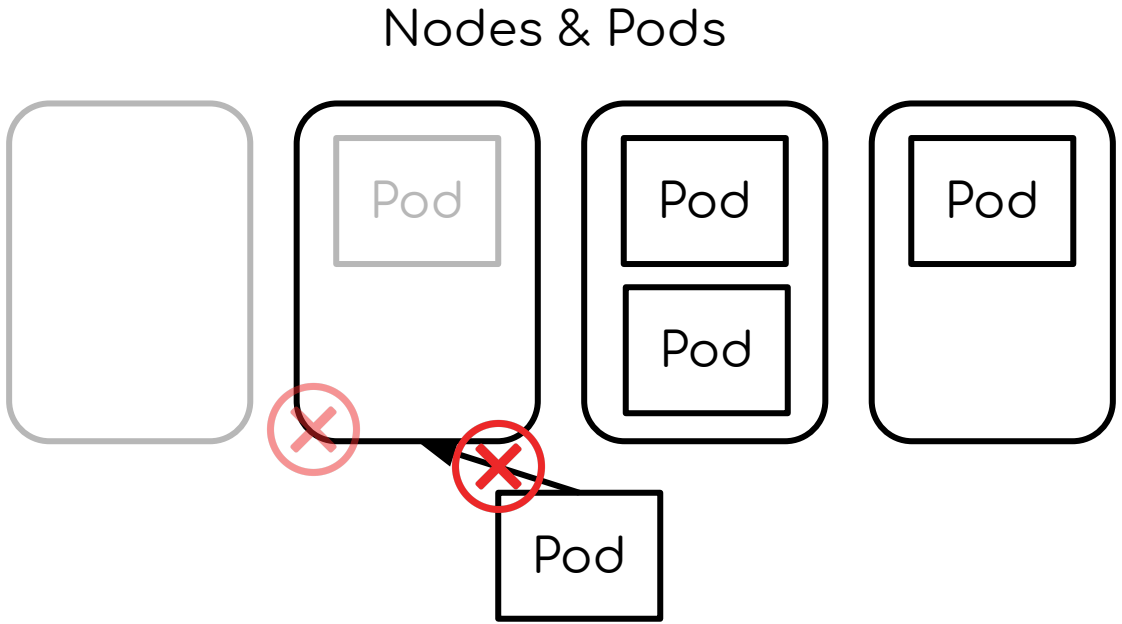
Unneeded nodes



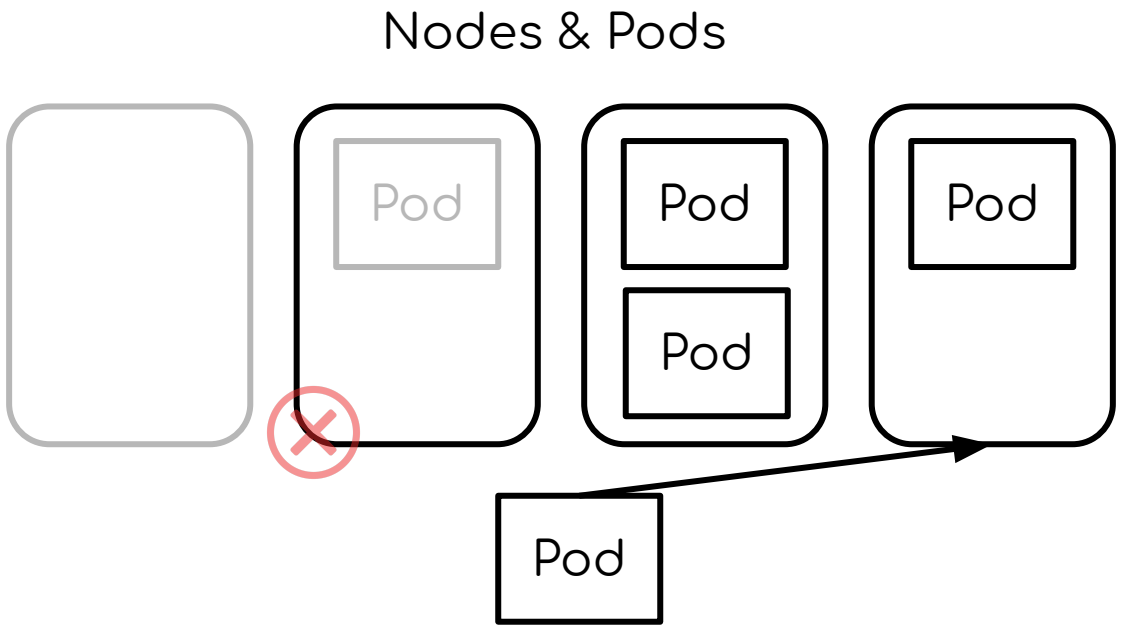
Unneeded nodes



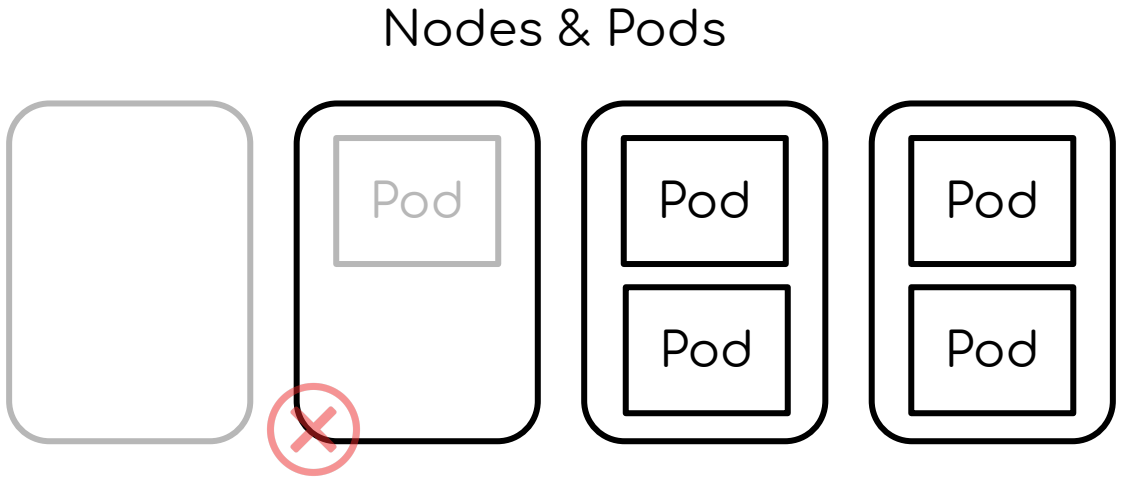
Unneeded nodes



Unneeded nodes

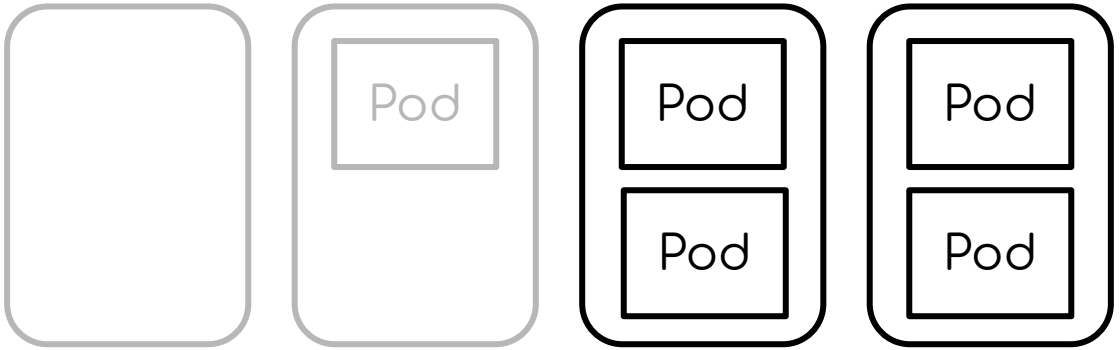


Unneeded nodes



Unneeded nodes

Nodes & Pods



Predict where pod can run

Naive solution:

check the pod's requests

Predict where pod can run

Naive solution:

check the pod's requests

check if pod tolerates node's taints

Predict where pod can run

Naive solution:

check the pod's requests

check if pod tolerates node's taints

check the pod's node selector

Predict where pod can run

Naive solution:

check the pod's requests

check if pod tolerates node's taints

check the pod's node selector

and affinity...

Predict where pod can run

Naive solution:

check the pod's requests

check if pod tolerates node's taints

check the pod's node selector

and affinity...

don't forget to account for host port conflicts

...

Predict where pod can run

Solution:

simulate scheduler's behavior by running default predicates

Predict where pod can run

Solution:

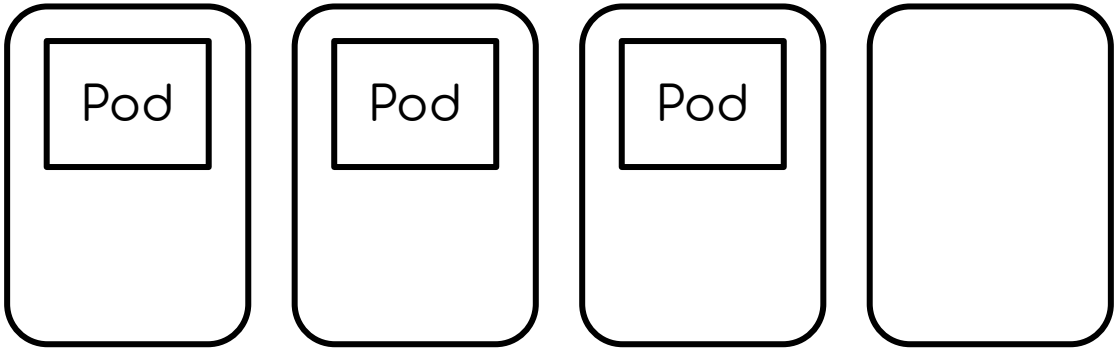
simulate scheduler's behavior by running default predicates

Caveat:

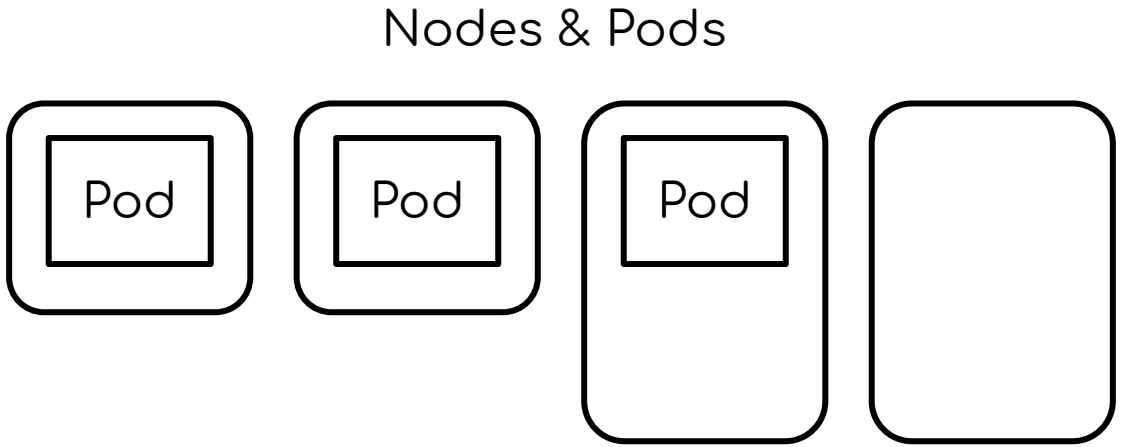
only supports fixed set of predicates

What will a new node look like?

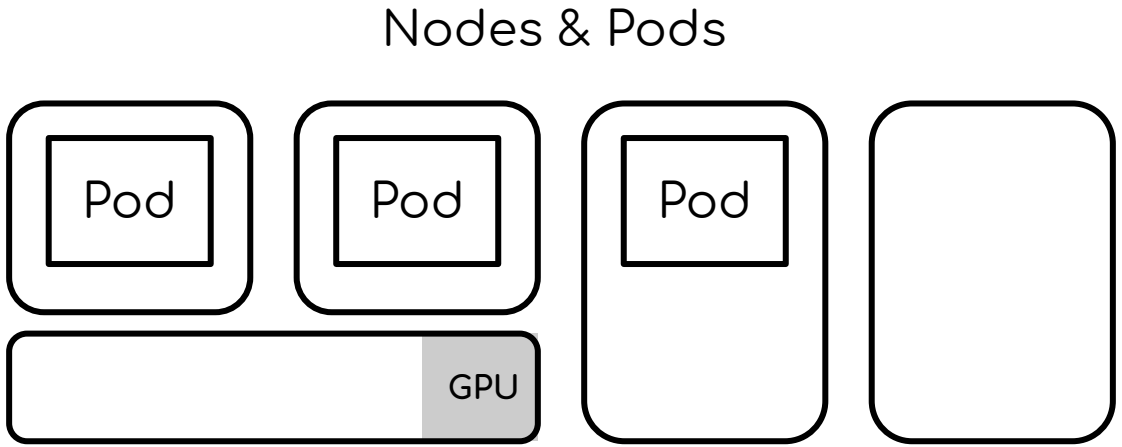
Nodes & Pods



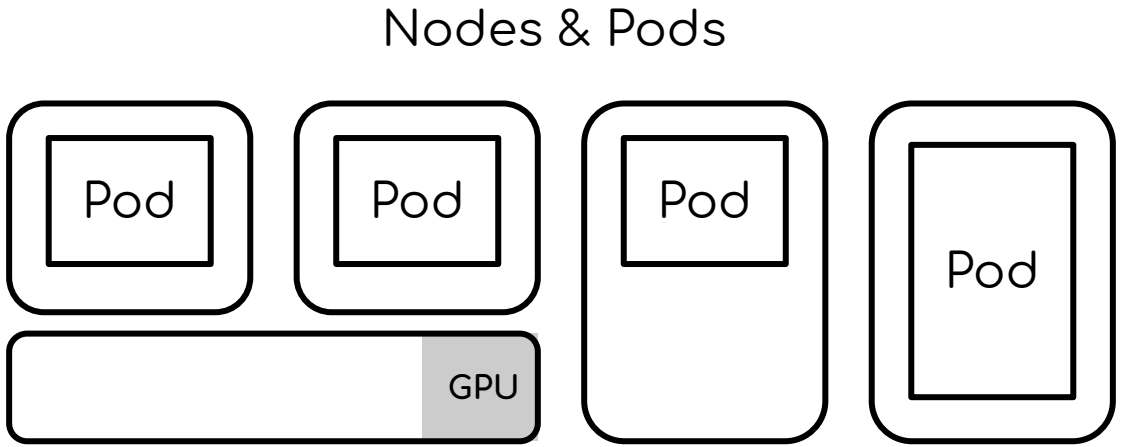
What will a new node look like?



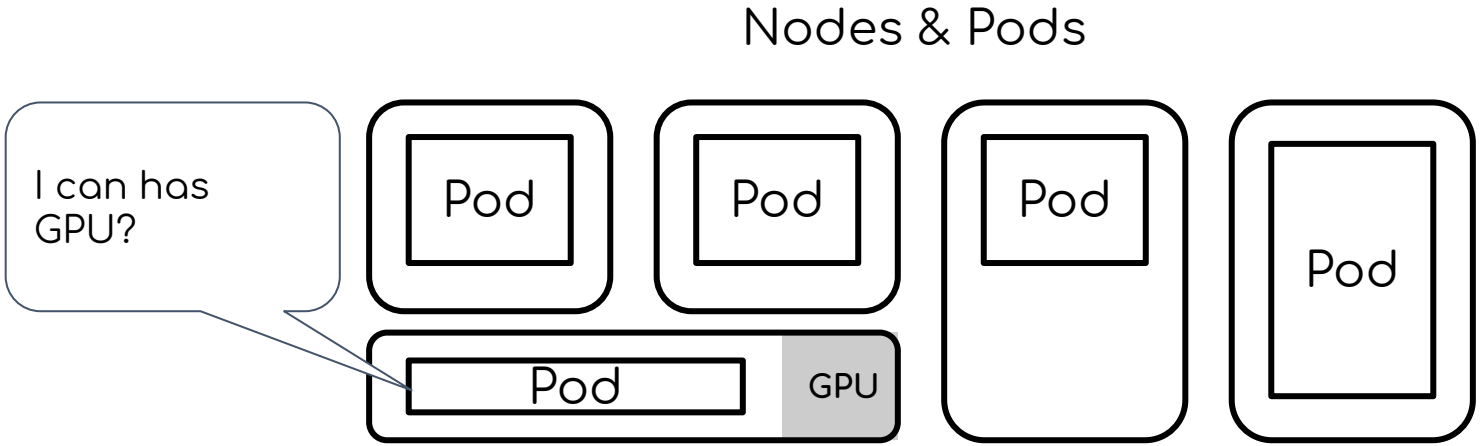
What will a new node look like?



What will a new node look like?

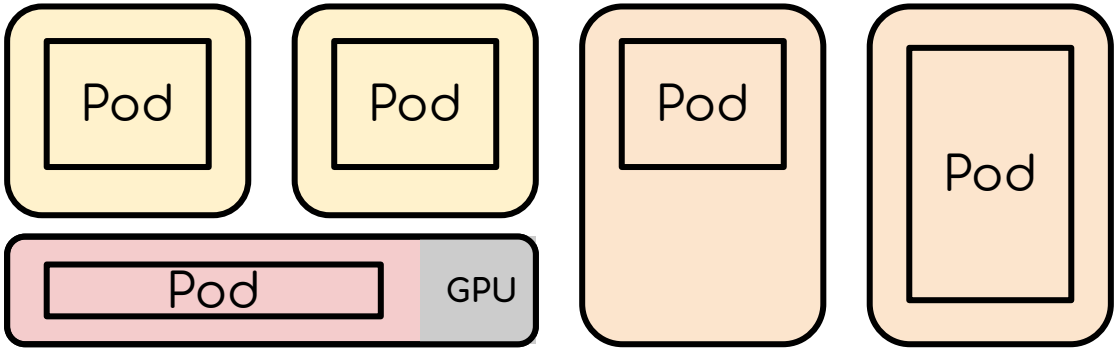


What will a new node look like?



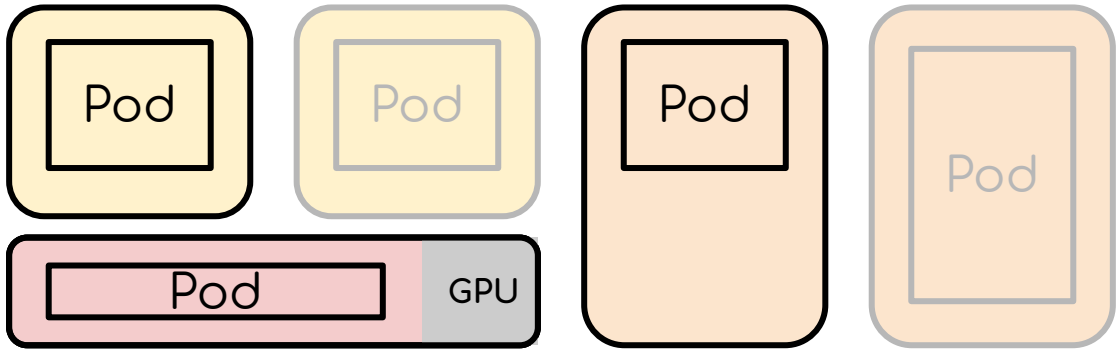
Node groups

Nodes & Pods



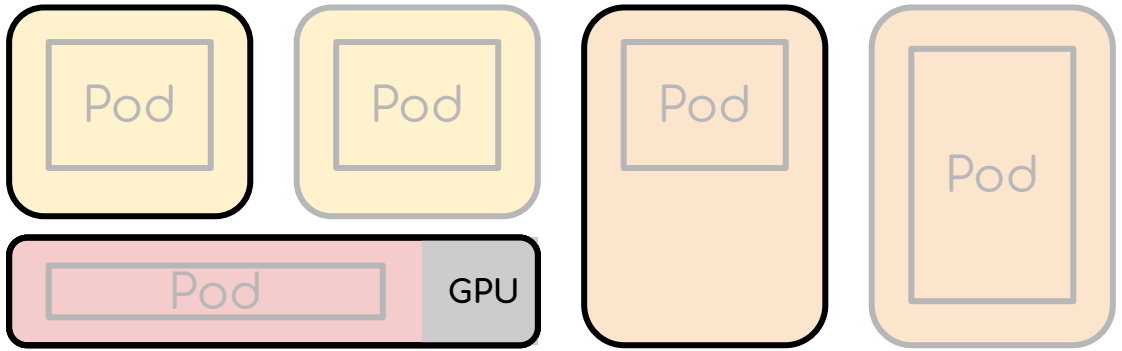
Node groups

Nodes & Pods



Node groups

Nodes & Pods



Cluster Autoscaler

What does CA do?

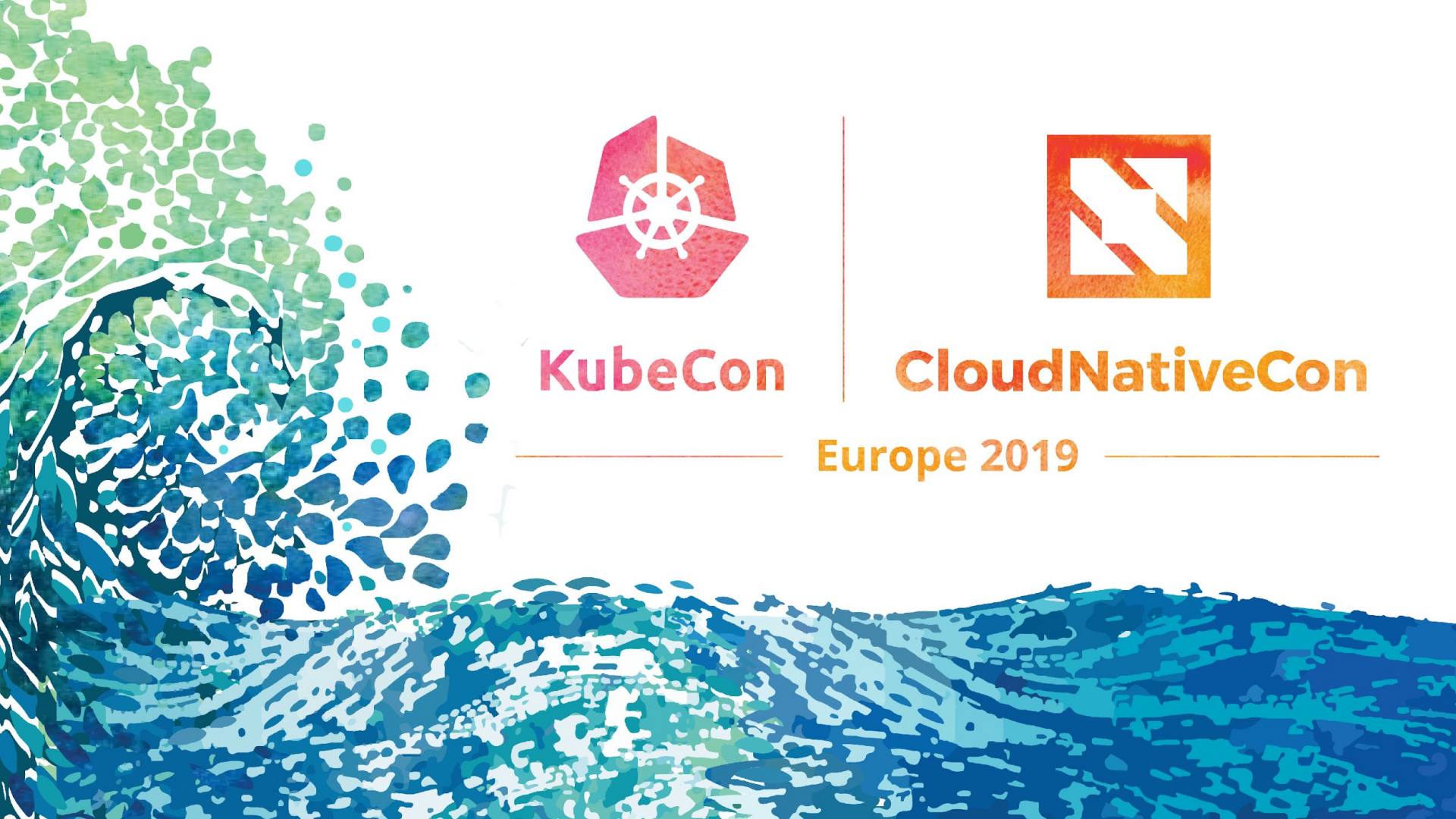
look for pending pods
simulate scheduler
add nodes by increasing node group size
delete particular nodes

Cluster Autoscaler

What doesn't CA do?

look at actual resource usage
register nodes in Kubernetes
configure nodes in any way
put any labels or taints on new nodes
support custom scheduling
predictive autoscaling


KubeCon
CloudNativeCon
Europe 2019
Questions?


KubeCon
CloudNativeCon
Europe 2019