

The Plesk logo is displayed in white lowercase letters on a dark blue background. The letter 'p' has a small horizontal line underneath it.

plesk

# Авто масштабируемый Kubernetes на AWS Spot'ах

Viktor Erpylev  
TechLead

# Ерпылёв Виктор

- Инженер
- 10+ лет в ИТ
- Ops -> DevOps -> TechLead



# Plesk

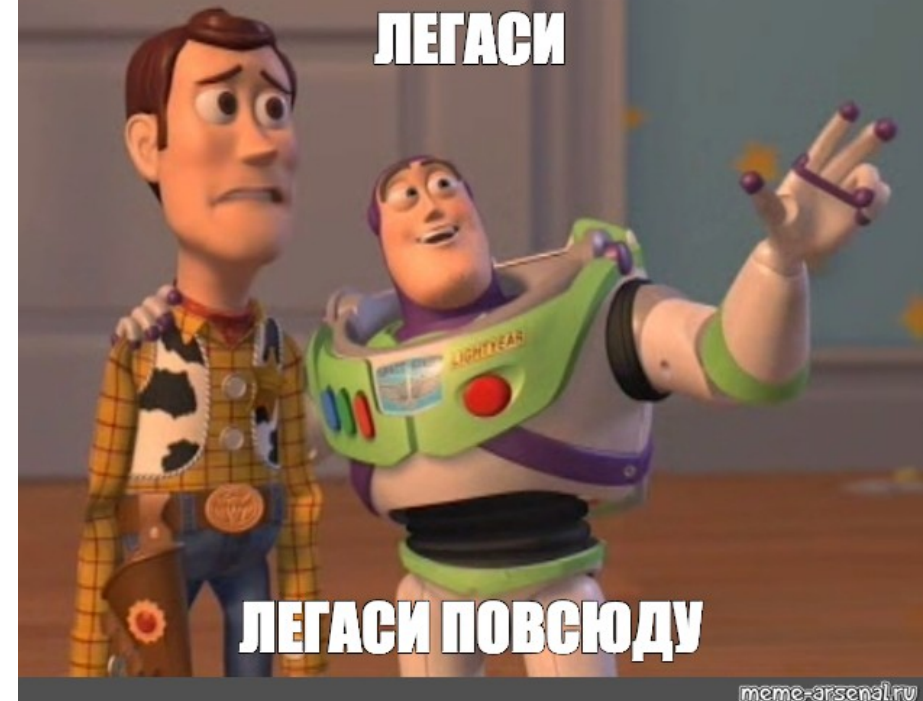
- 400k инсталляций продукта
- 6% сайтов в сети интернет
- Больше 20 лет
- 120+ классных инженеров

plesk

**НЕ ЛЕГАСИ**



**ЛЕГАСИ**



**ЛЕГАСИ ПОВСЮДУ**

memе-arsenal.ru

**А ЛЕГАСИЩЕ!**

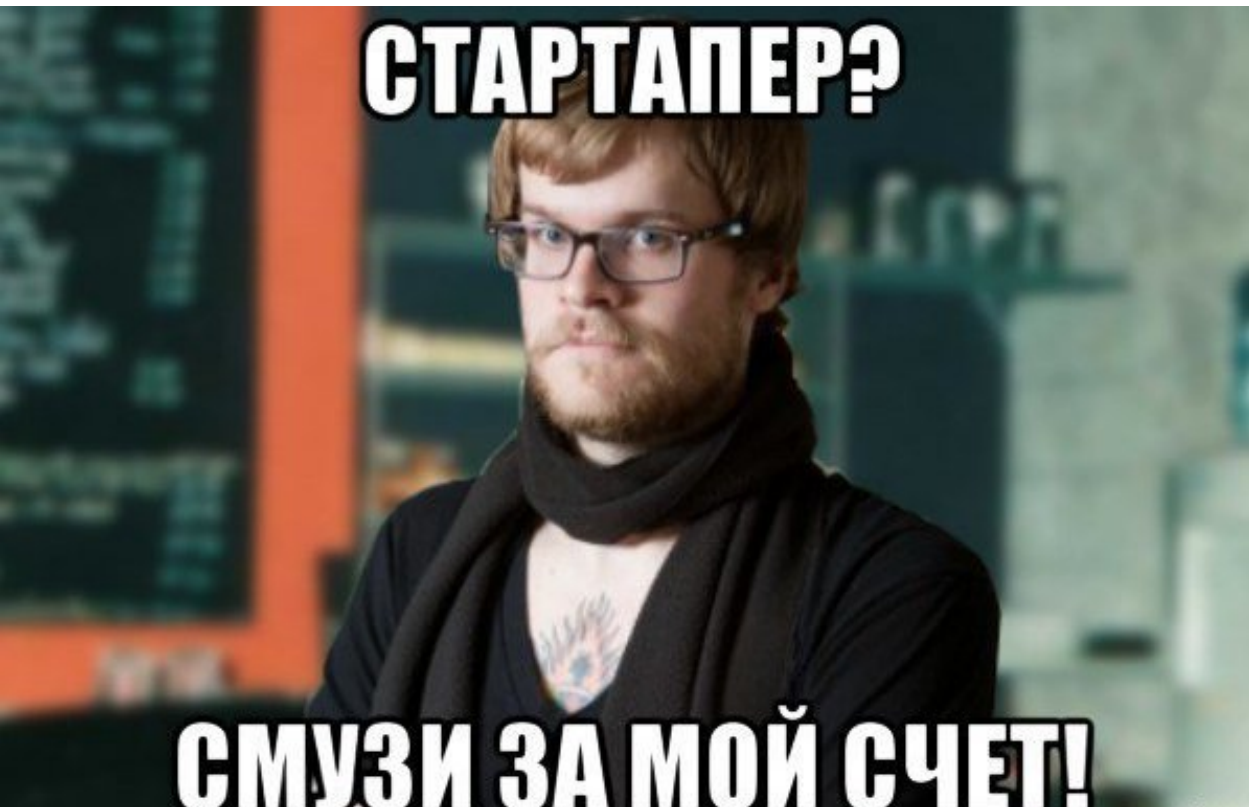


МТС

memе-arsenal.ru

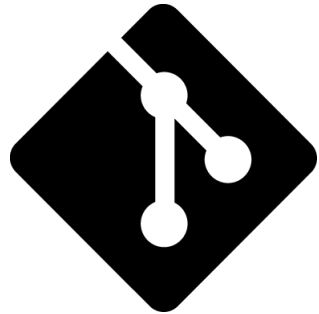
\*Тут возможно будет один, но более подходящий мем про легаси\*



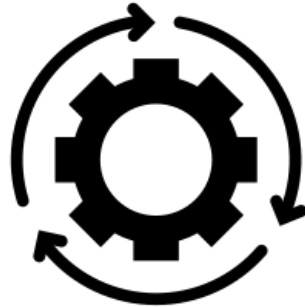
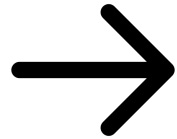


Какой стартап без k8s?

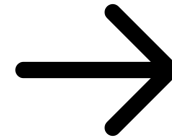
# Web Service Platform



Code



Build



Deploy

# Web Service Platform



Фреймворк для создания CI/CD и запуска асинхронных задач в k8s

Сборка Docker образов в k8s кластере



## Terraform

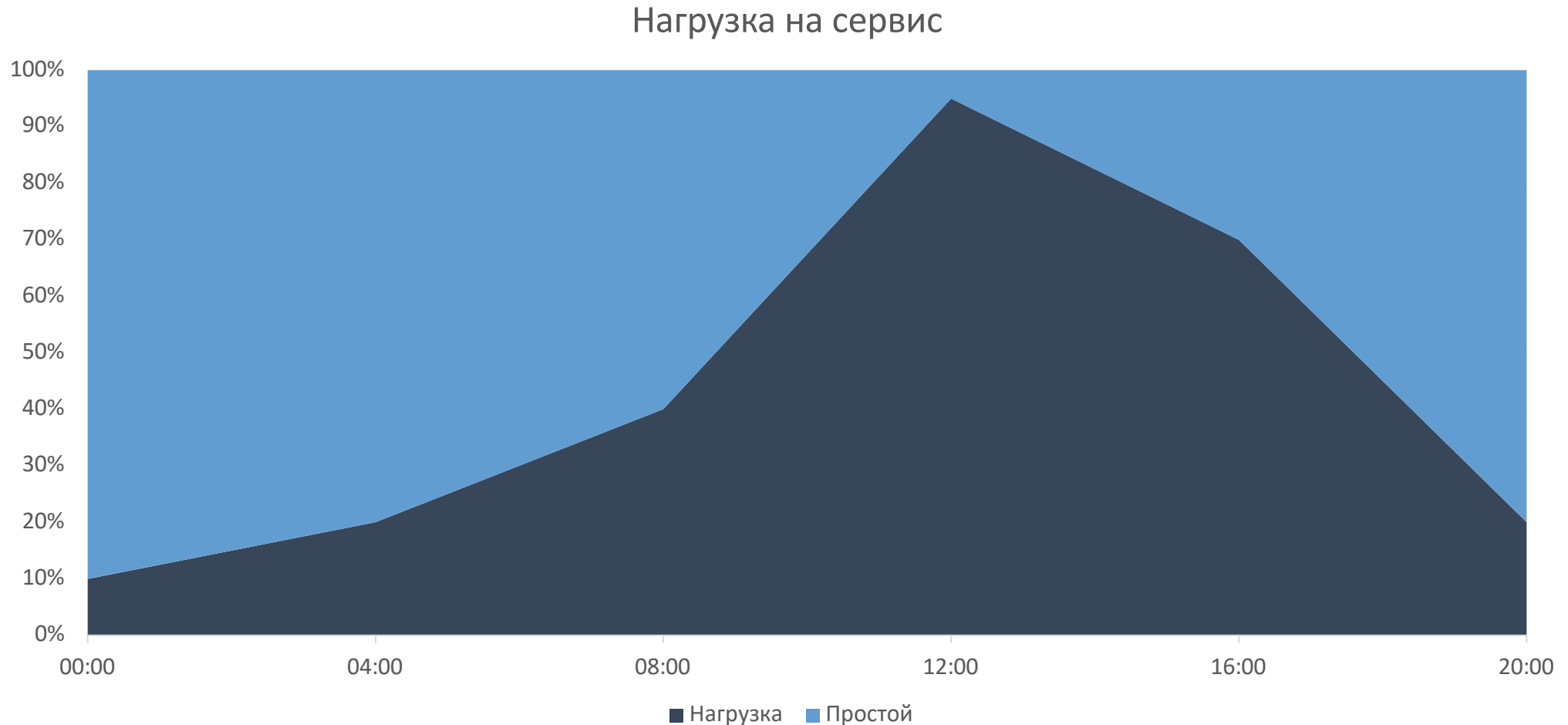
Инструмент для управления инфраструктурой в облаке

# Виды нагрузки в кластере

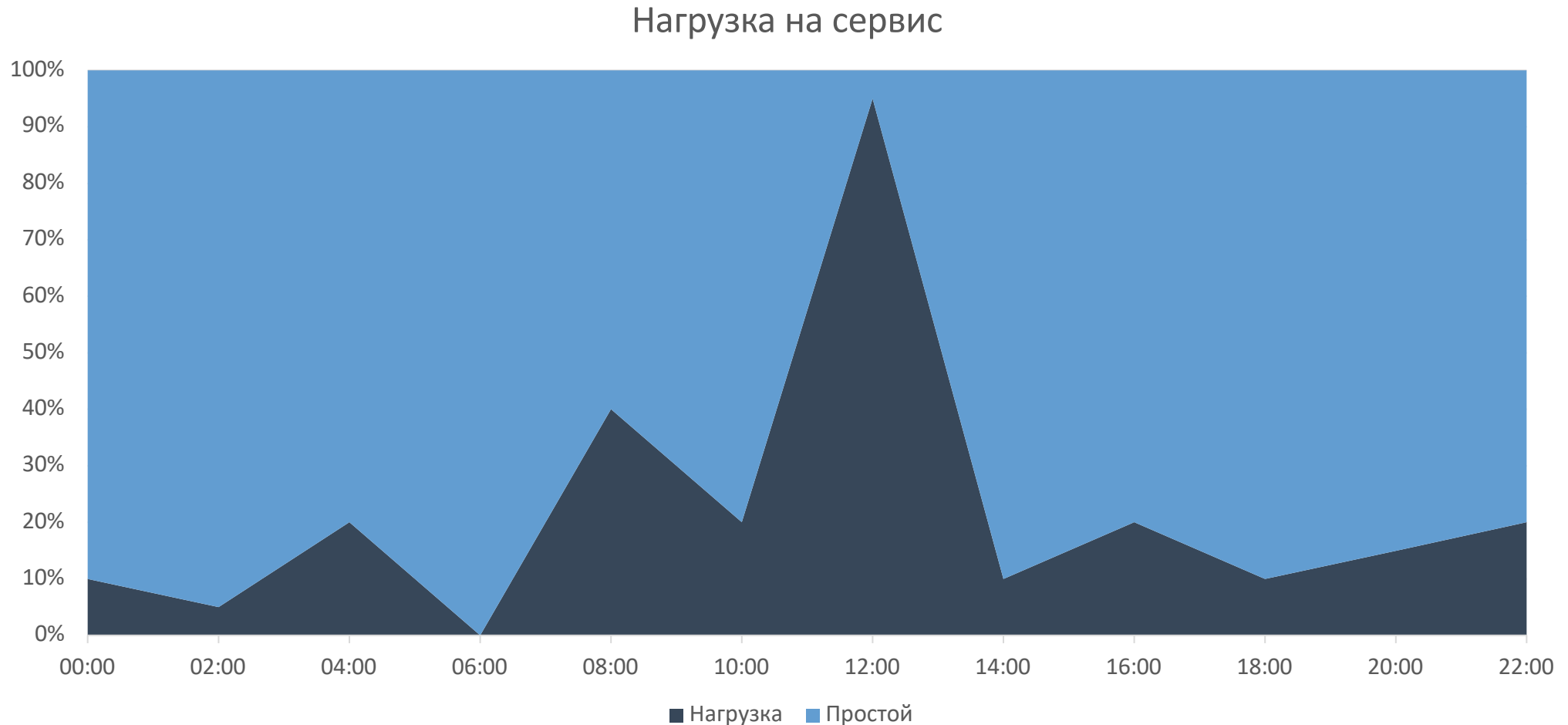
- Наша нагрузка
  - Контрольная панель
  - Мониторинг
- Клиентская нагрузка
  - Сборки Docker образов
  - Развертывание инфраструктуры



# Как быть, если нагрузка не прогнозируется?



# Как быть, если нагрузка не прогнозируется?



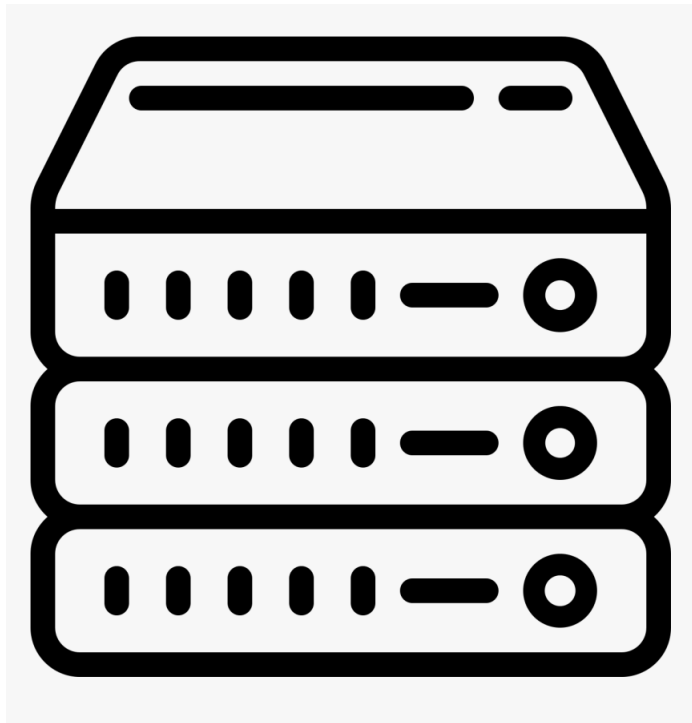
# Решение проблемы

An iceberg floating in the ocean. The top part is visible above the water, and the much larger bottom part is submerged. The sky is blue with some clouds, and the water is a deep blue.

Контекст, нужный  
для понимания

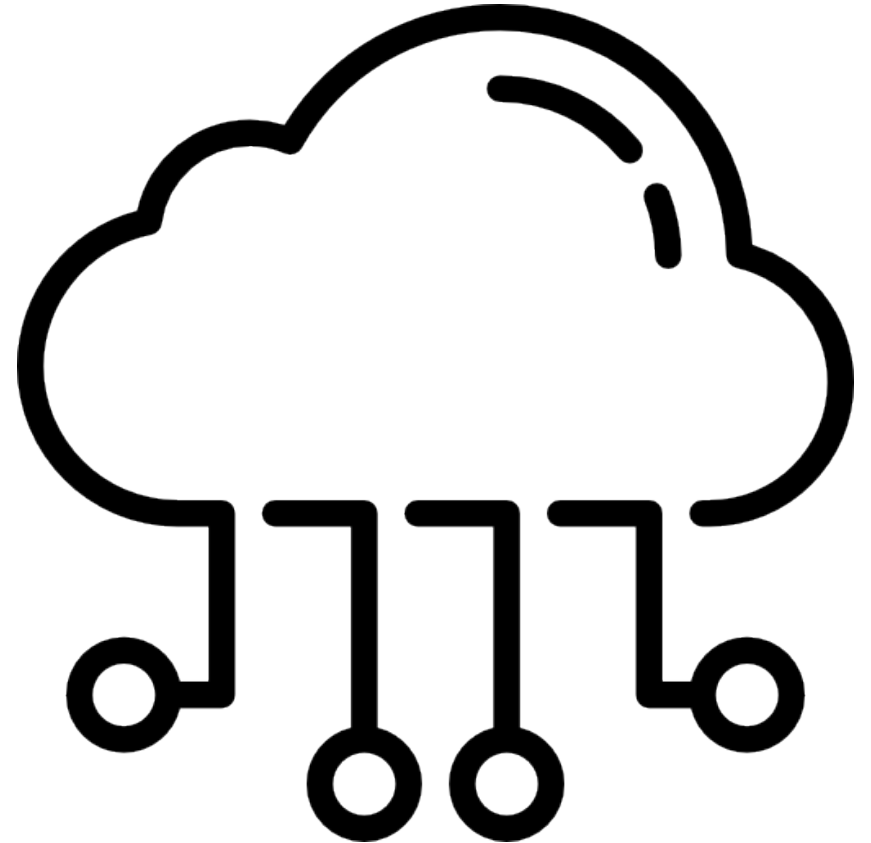
# Зачем свой kubernetes?

- Одинаковое решение на bare metal и в облаке

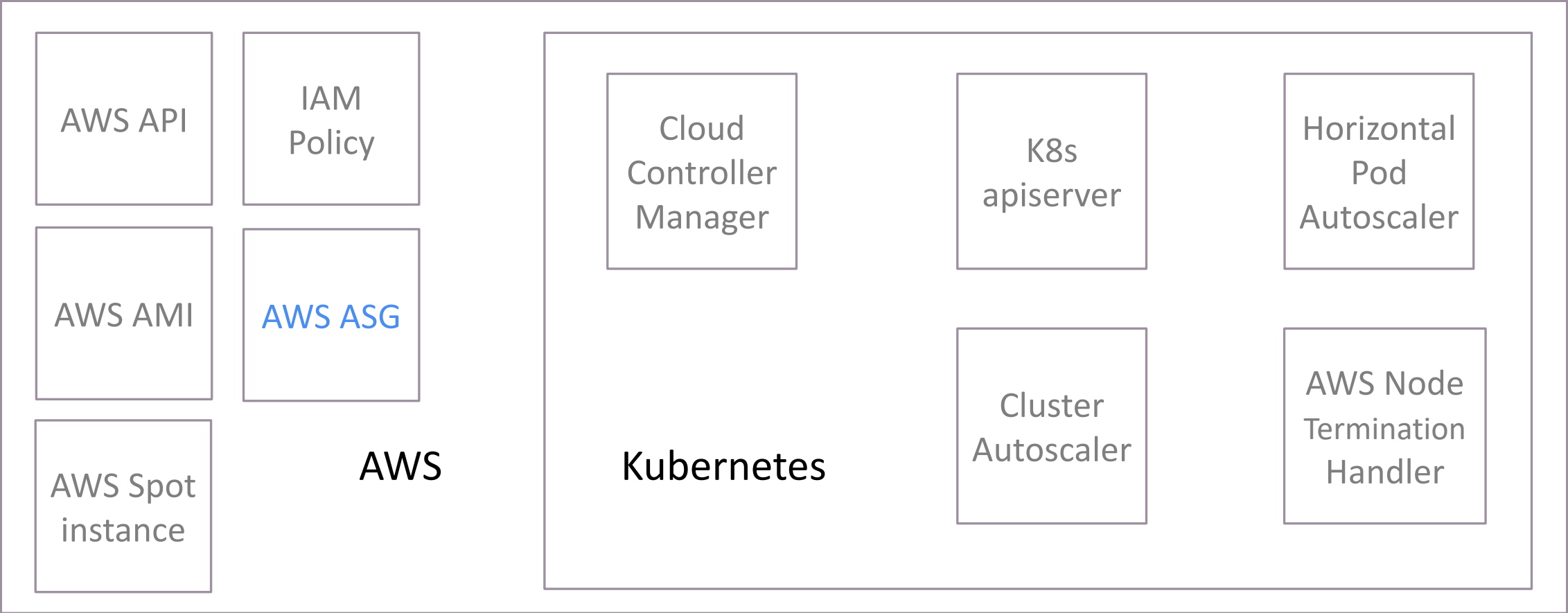


# Cluster autoscaler

- GCE
- GKE
- AWS
- Azure
- Alibaba Cloud
- OpenStack  
Magnum
- DigitalOcean
- CloudStack
- Exoscale
- Packet

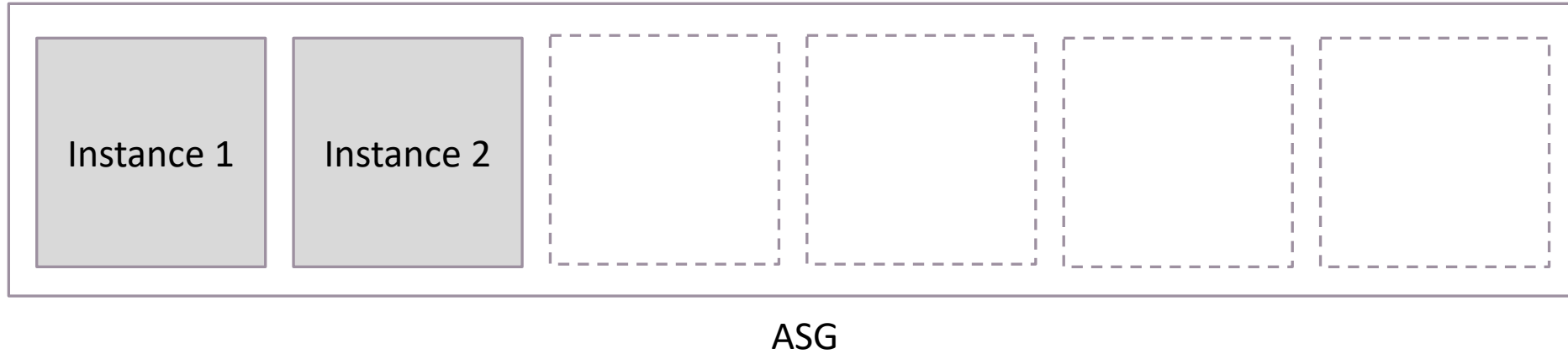


# Архитектура решения

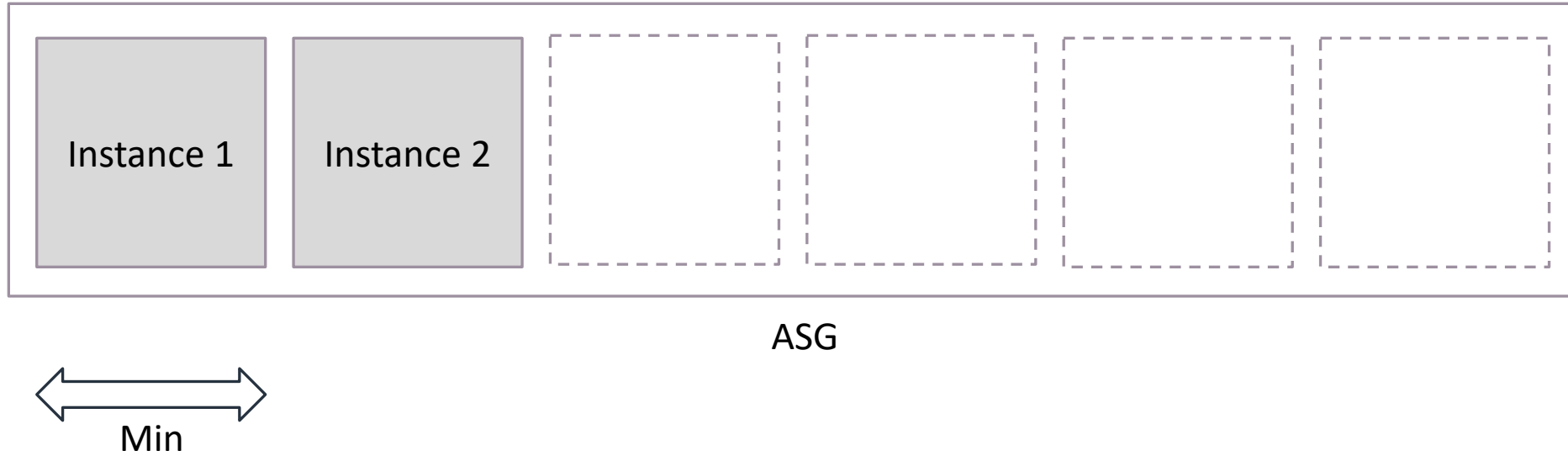




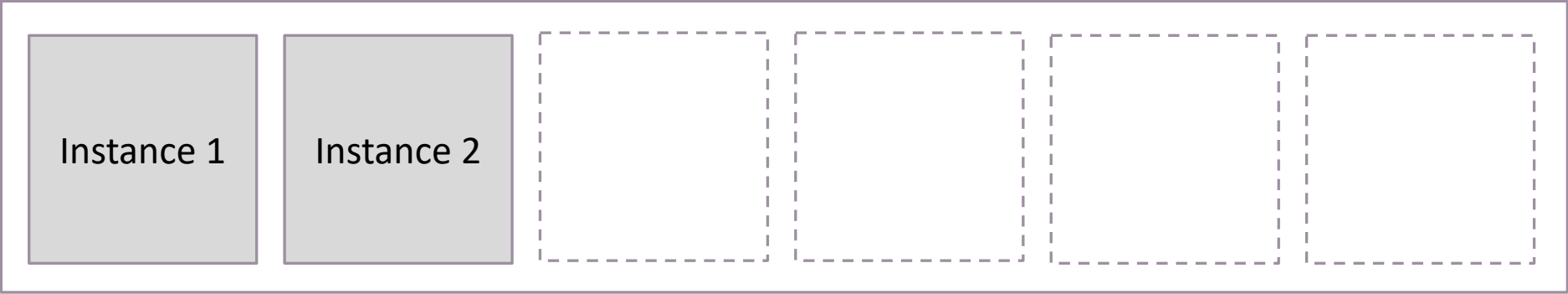
# AWS – Auto Scaling groups



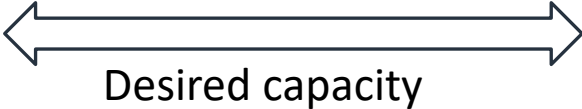
# AWS – Auto Scaling groups



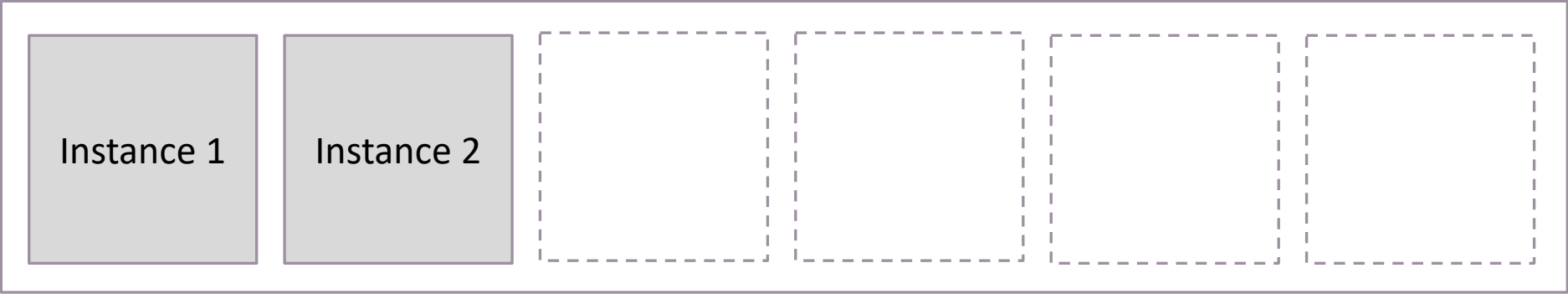
# AWS – Auto Scaling groups



ASG



# AWS – Auto Scaling groups



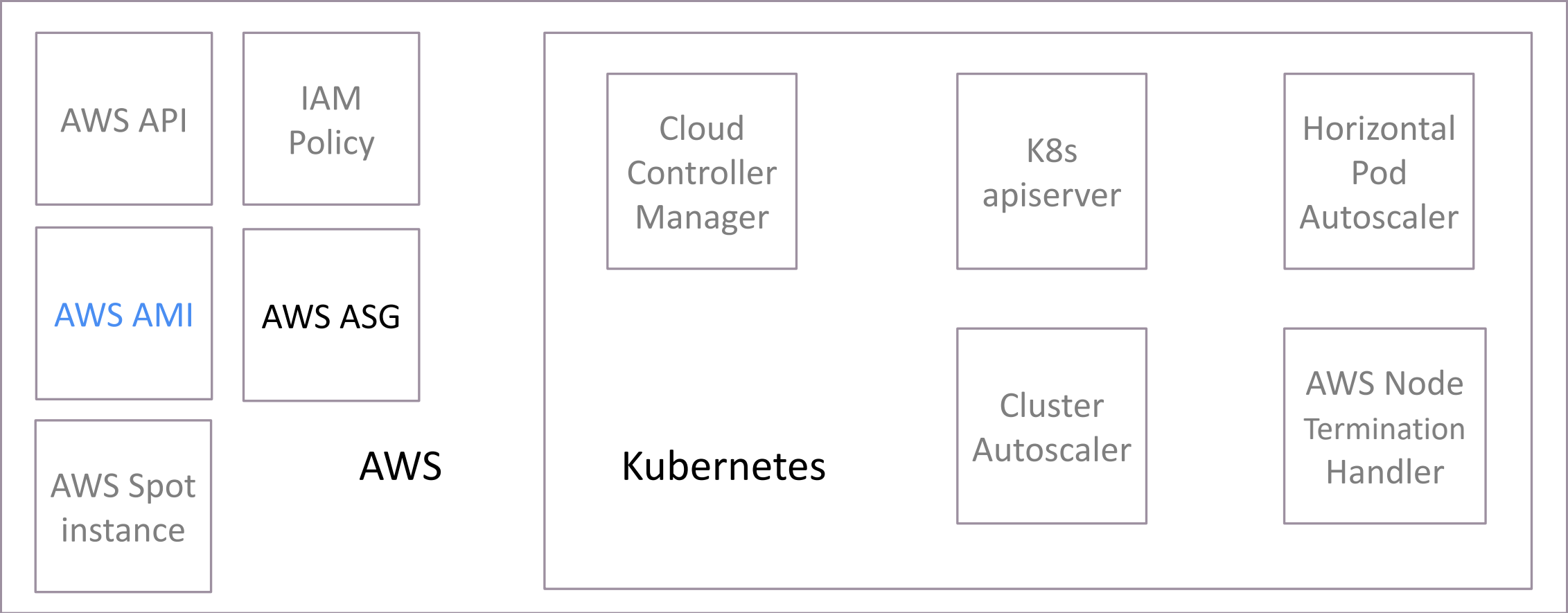
ASG



# AWS – Auto Scaling groups

- Поддержка текущего количества машин
- Масштабирование
  - По расписанию
  - По запросу
  - Вручную

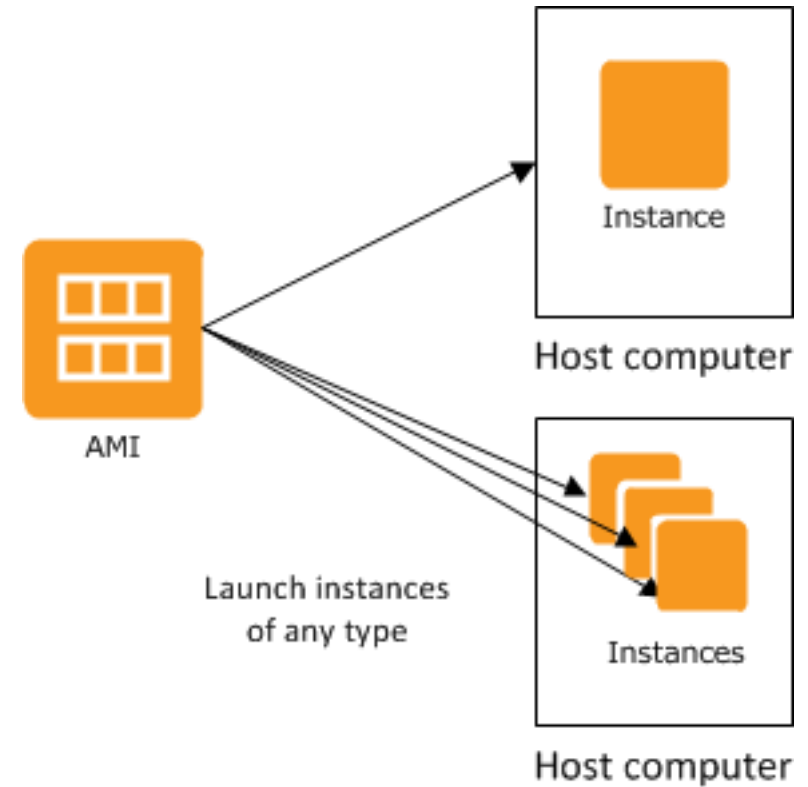
# Архитектура решения



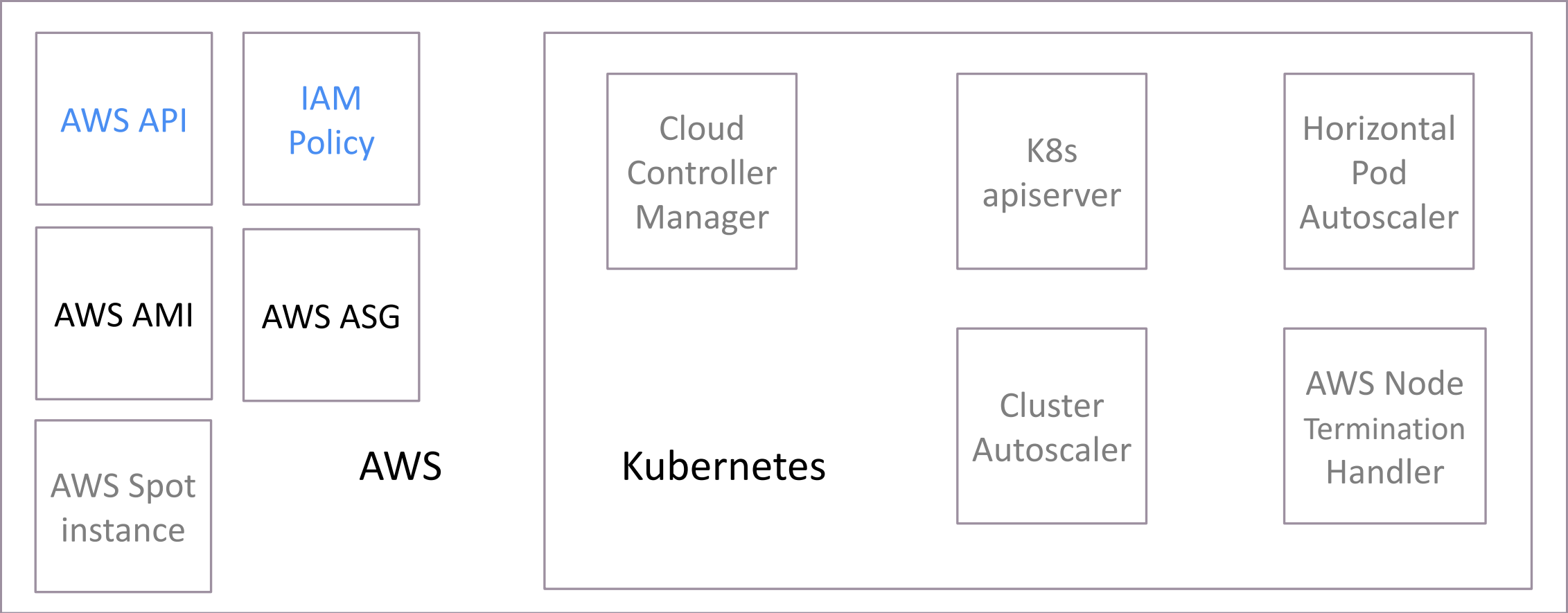


# AWS – Amazon Machine Image

- Предподготовленный Centos 8
  - Containerd
  - Kubeadm
  - Kubelet
- Launch Template
  - Join token
  - Node labels



# Архитектура решения



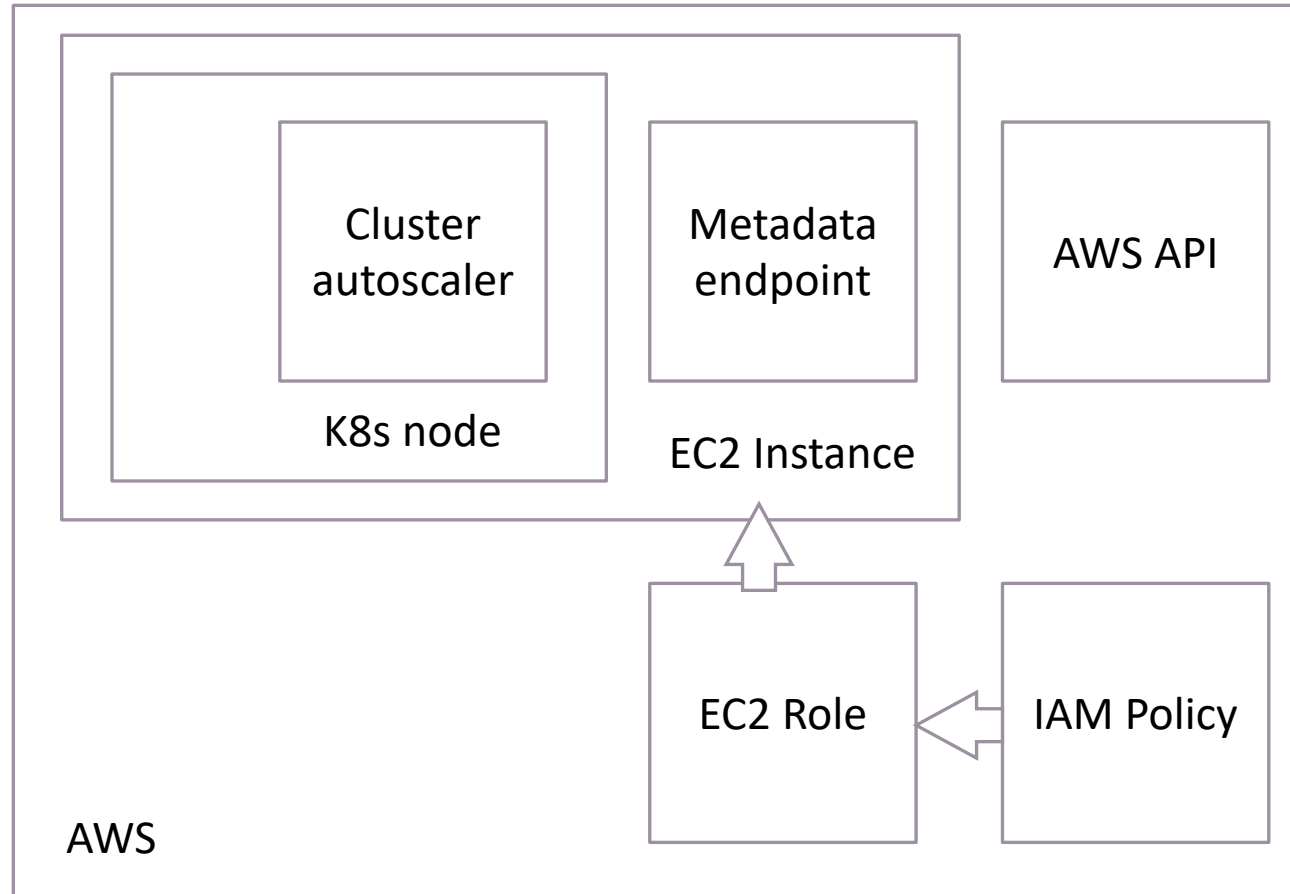
# IAM policy

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "autoscaling:DescribeAutoScalingGroups",
        "autoscaling:DescribeAutoScalingInstances",
        "autoscaling:DescribeLaunchConfigurations",
        "autoscaling:DescribeTags",
        "autoscaling:SetDesiredCapacity",
        "autoscaling:TerminateInstanceInAutoScalingGroup"
      ],
      "Resource": "*"
    }
  ]
}
```

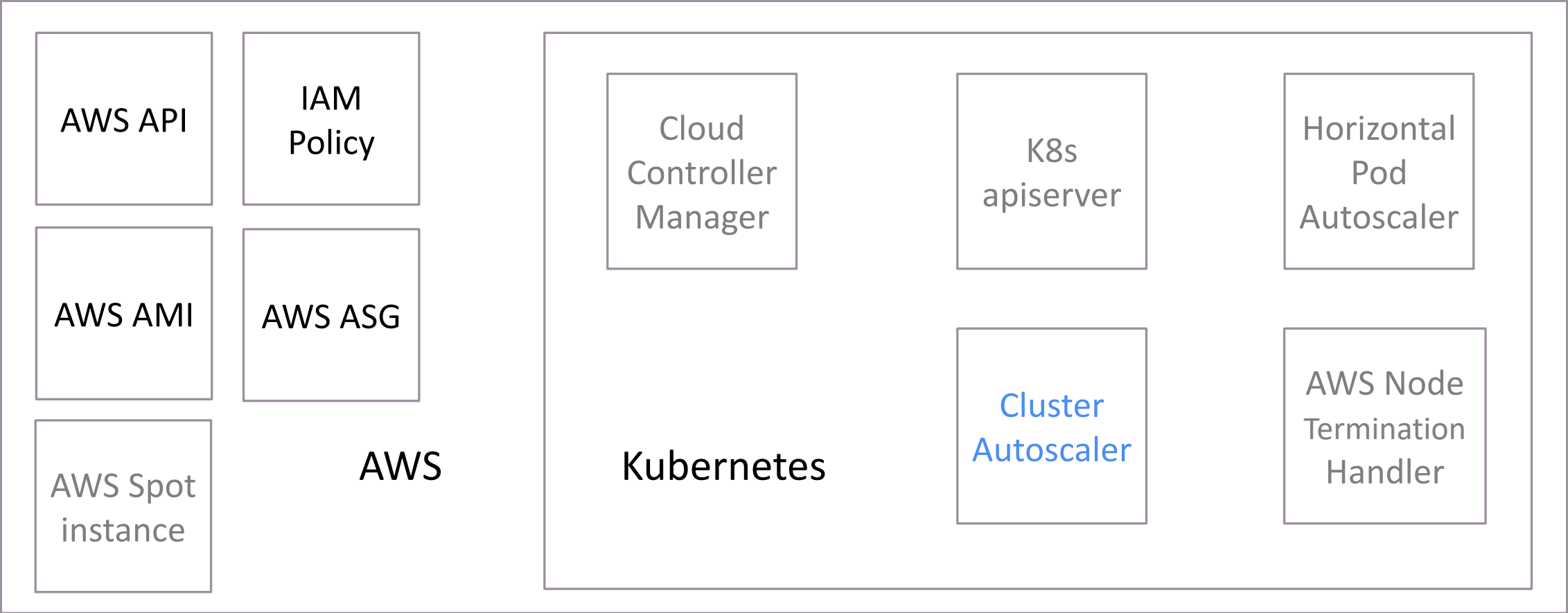
\*Тут есть проблема

Для того, чтобы cluster autoscaler мог добавлять и убавлять ноды ему нужны права в AWS

# AWS API access



# Архитектура решения



# Cluster autoscaler

Меняет размер кластера  
события:

- Появился под, которому не хватает ресурсов
- Нода может быть освобождена





# Cluster autoscaler— это просто

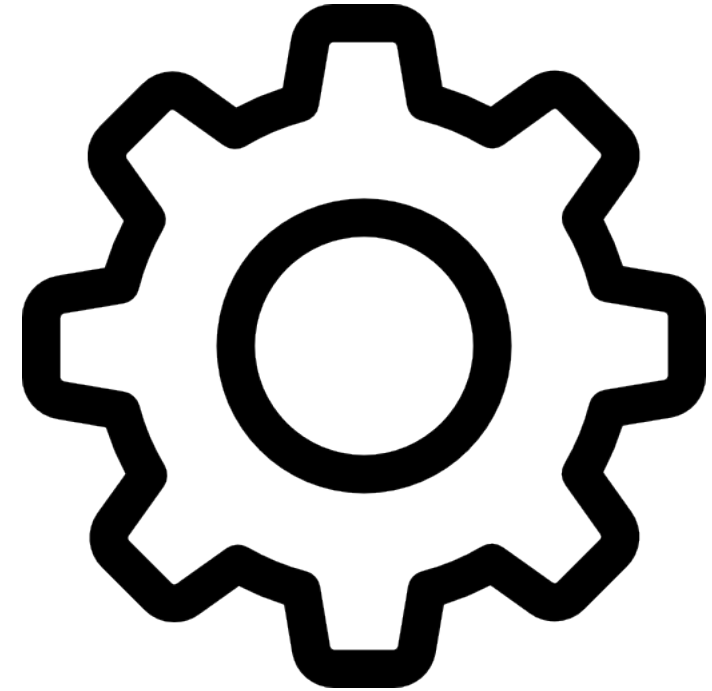
Parameter	Description	Default
cluster-name	Autoscaled cluster name, if available	""
address	The address to expose prometheus metrics	:8085
kubernetes	Kubernetes master location. Leave blank for default	""
kubeconfig	Path to kubeconfig file with authorization and master location information	""
cloud-config	The path to the cloud provider configuration file. Empty string for no configuration file	""
namespace	Namespace in which cluster-autoscaler run	"kube-system"
scale-down-enabled	Should CA scale down the cluster	true
scale-down-delay-after-add	How long after scale up that scale down evaluation resumes	10 minutes
scale-down-delay-after-delete	How long after node deletion that scale down evaluation resumes, defaults to scan-interval	scan-interval
scale-down-delay-after-failure	How long after scale down failure that scale down evaluation resumes	3 minutes
scale-down-unneded-time	How long a node should be unneeded before it is eligible for scale down	10 minutes
scale-down-unready-time	How long an unready node should be unneeded before it is eligible for scale down	20 minutes
scale-down-utilization-threshold	Node utilization level, defined as sum of requested resources divided by capacity, below which a node can be considered for scale down	0.5
scale-down-non-empty-candidates-count	Maximum number of non empty nodes considered in one iteration as candidates for scale down with drain Lower value means better CA responsiveness but possible slower scale down latency Higher value can affect CA performance with big clusters (hundreds of nodes) Set to non positive value to turn this heuristic off - CA will not limit the number of nodes it considers."	30
scale-down-candidates-pool-ratio	A ratio of nodes that are considered as additional non empty candidates for scale down when some candidates from previous iteration are no longer valid Lower value means better CA responsiveness but possible slower scale down latency Higher value can affect CA performance with big clusters (hundreds of nodes) Set to 1.0 to turn this heuristics off - CA will take all nodes as additional candidates.	0.1

scale-down-candidates-pool-min-count	Minimum number of nodes that are considered as additional non empty candidates for scale down when some candidates from previous iteration are no longer valid. When calculating the pool size for additional candidates we take $\max(\#nodes * scale-down-candidates-pool-ratio, scale-down-candidates-pool-min-count)$	50
scan-interval	How often cluster is reevaluated for scale up or down	10 seconds
max-nodes-total	Maximum number of nodes in all node groups. Cluster autoscaler will not grow the cluster beyond this number.	0
cores-total	Minimum and maximum number of cores in cluster, in the format <code>min-max</code> . Cluster autoscaler will not scale the cluster beyond these numbers.	320000
memory-total	Minimum and maximum number of gigabytes of memory in cluster, in the format <code>min-max</code> . Cluster autoscaler will not scale the cluster beyond these numbers.	6400000
gpu-total	Minimum and maximum number of different GPUs in cluster, in the format <code>&lt;gpu_type&gt;::min-max</code> . Cluster autoscaler will not scale the cluster beyond these numbers. Can be passed multiple times. CURRENTLY THIS FLAG ONLY WORKS ON GKE.	""
cloud-provider	Cloud provider type.	gce
max-empty-bulk-delete	Maximum number of empty nodes that can be deleted at the same time.	10
max-graceful-termination-sec	Maximum number of seconds CA waits for pod termination when trying to scale down a node.	600
max-total-unready-percentage	Maximum percentage of unready nodes in the cluster. After this is exceeded, CA halts operations	45
ok-total-unready-count	Number of allowed unready nodes, irrespective of max-total-unready-percentage	3
max-node-provisioning-time	Maximum time CA waits for node to be provisioned	15 minutes
nodes	sets min,max size and other configuration data for a node group in a format accepted by cloud provider. Can be used multiple times. Format: <code>::&lt;other...&gt;</code>	""
node-group-auto-discovery	One or more definition(s) of node group auto-discovery. A definition is expressed <code>&lt;name of discoverer&gt;:[&lt;key&gt;[=&lt;value&gt;]]</code> The <code>aws</code> , <code>gce</code> , and <code>azure</code> cloud providers are currently supported. AWS matches by ASG tags, e.g. <code>asg:tag=tagKey,anotherTagKey</code> GCE matches by IG name prefix, and requires you to specify min and max nodes per IG, e.g. <code>mig:namePrefix=pfx,min=0,max=10</code> Azure matches by tags on VMSS, e.g. <code>label:foo=bar</code> , and will auto-detect <code>min</code> and <code>max</code> tags on the VMSS to set scaling limits. Can be used multiple times	""

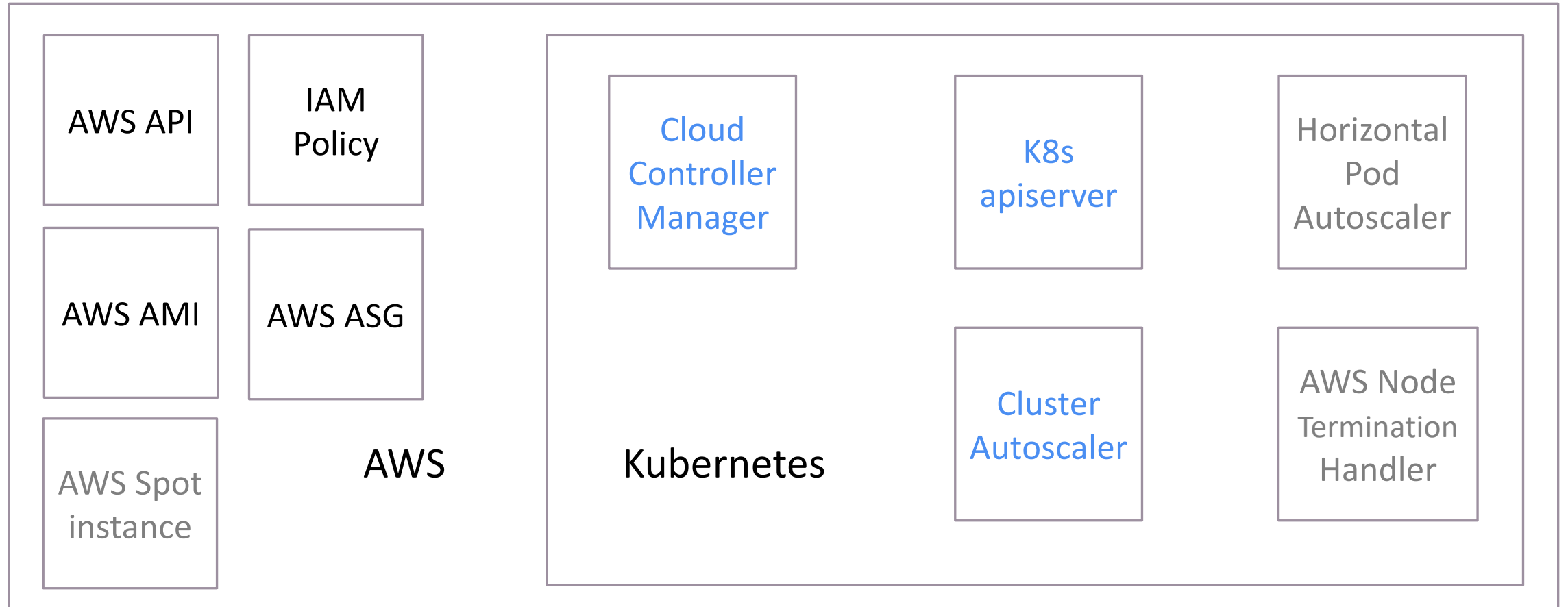
estimator	Type of resource estimator to be used in scale up	binpacking
expander	Type of node group expander to be used in scale up.	random
write-status-configmap	Should CA write status information to a configmap	true
max-inactivity	Maximum time from last recorded autoscaler activity before automatic restart	10 minutes
max-failing-time	Maximum time from last recorded successful autoscaler run before automatic restart	15 minutes
balance-similar-node-groups	Detect similar node groups and balance the number of nodes between them	false
balancing-ignore-label	Define a node label that should be ignored when considering node group similarity. One label per flag occurrence.	""
node-autoprovisioning-enabled	Should CA autoprovision node groups when needed	false
max-autoprovisioned-node-group-count	The maximum number of autoprovisioned groups in the cluster	15
unremovable-node-recheck-timeout	The timeout before we check again a node that couldn't be removed before	5 minutes
expendable-pods-priority-cutoff	Pods with priority below cutoff will be expendable. They can be killed without any consideration during scale down and they don't cause scale up. Pods with null priority (PodPriority disabled) are non expendable	-10
regional	Cluster is regional	false
leader-elect	Start a leader election client and gain leadership before executing the main loop. Enable this when running replicated components for high availability	true
leader-elect-lease-duration	The duration that non-leader candidates will wait after observing a leadership renewal until attempting to acquire leadership of a led but unrenewed leader slot. This is effectively the maximum duration that a leader can be stopped before it is replaced by another candidate. This is only applicable if leader election is enabled	15 seconds
leader-elect-renew-deadline	The interval between attempts by the acting master to renew a leadership slot before it stops leading. This must be less than or equal to the lease duration. This is only applicable if leader election is enabled	10 seconds
leader-elect-retry-period	The duration the clients should wait between attempting acquisition and renewal of a leadership. This is only applicable if leader election is enabled	2 seconds
leader-elect-retry-period	The duration the clients should wait between attempting acquisition and renewal of a leadership. This is only applicable if leader election is enabled	2 seconds
leader-elect-resource-lock	The type of resource object that is used for locking during leader election. Supported options are <code>endpoints</code> (default) and <code>configmaps</code>	"endpoints"
aws-use-static-instance-list	Should CA fetch instance types in runtime or use a static list. AWS only	false
skip-nodes-with-system-pods	If true cluster autoscaler will never delete nodes with pods from kube-system (except for DaemonSet or mirror pods)	true
skip-nodes-with-local-storage	If true cluster autoscaler will never delete nodes with pods with local storage, e.g. EmptyDir or HostPath	true
min-replica-count	Minimum number or replicas that a replica set or replication controller should have to allow their pods deletion in scale down	0

# Конфигурация

- skip-nodes-with-local-storage
- node-group-auto-discovery



# Архитектура решения



# Cloud controller manager

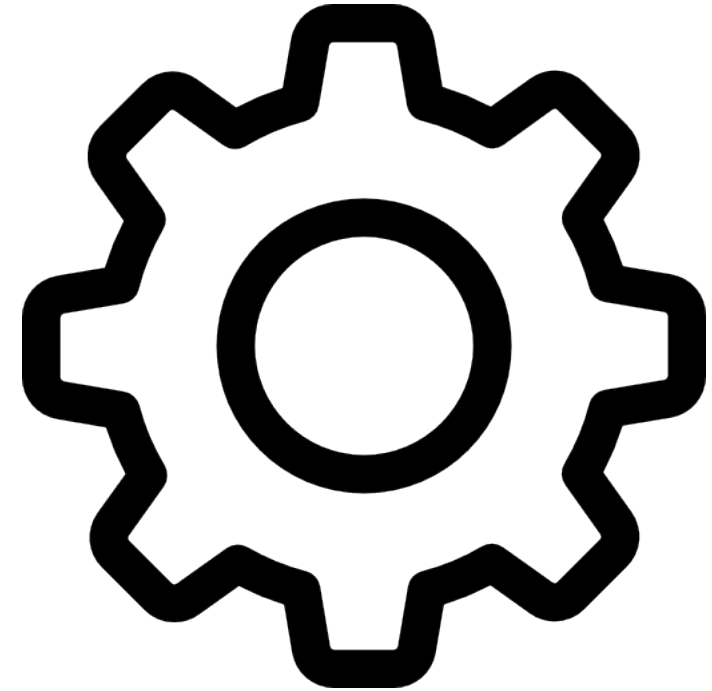
- Часть Cloud Provider
- Забирает из AWS API информацию по инстансам и обогащает лейблы ноды
  - instance id
  - region

# Конфигурация

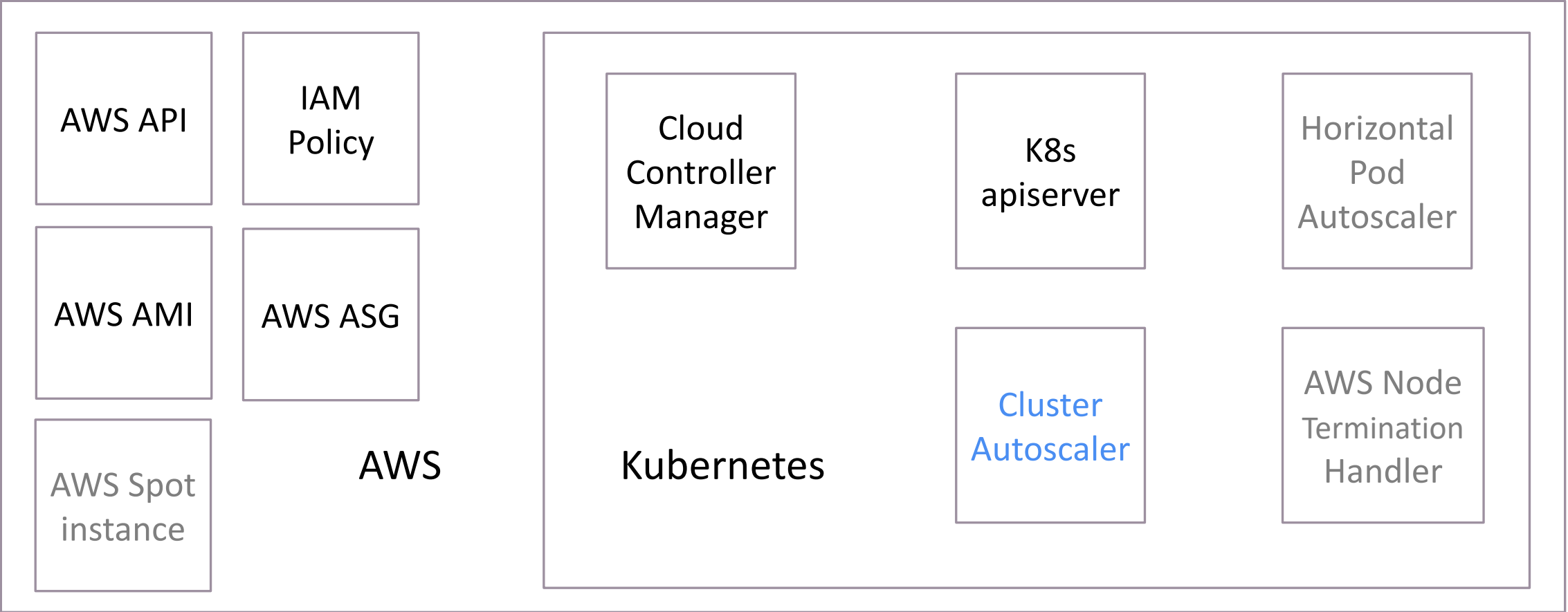
Cluster autoscaler

сопоставляет AWS Resource

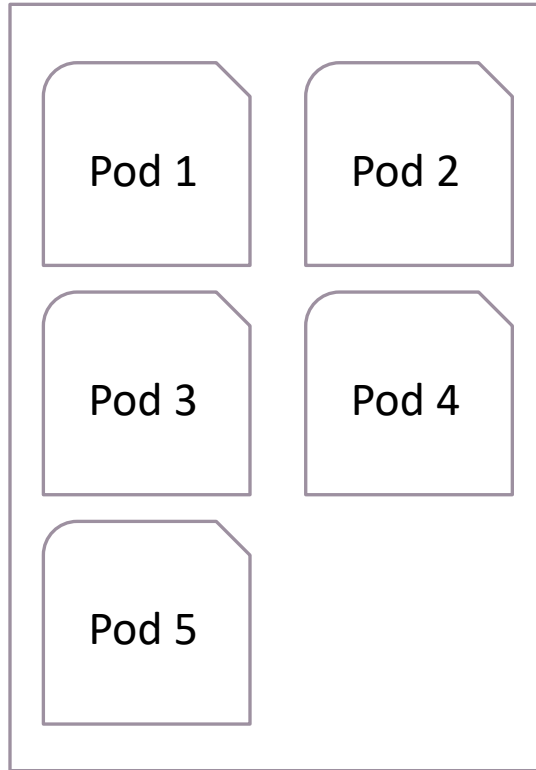
tags и k8s node labels



# Архитектура решения

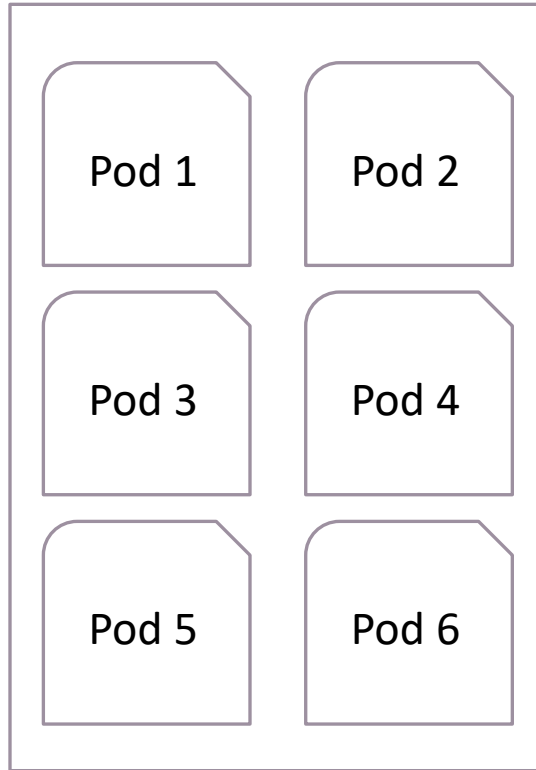


# Как работает Cluster Autoscaler: Upscale



Node 1

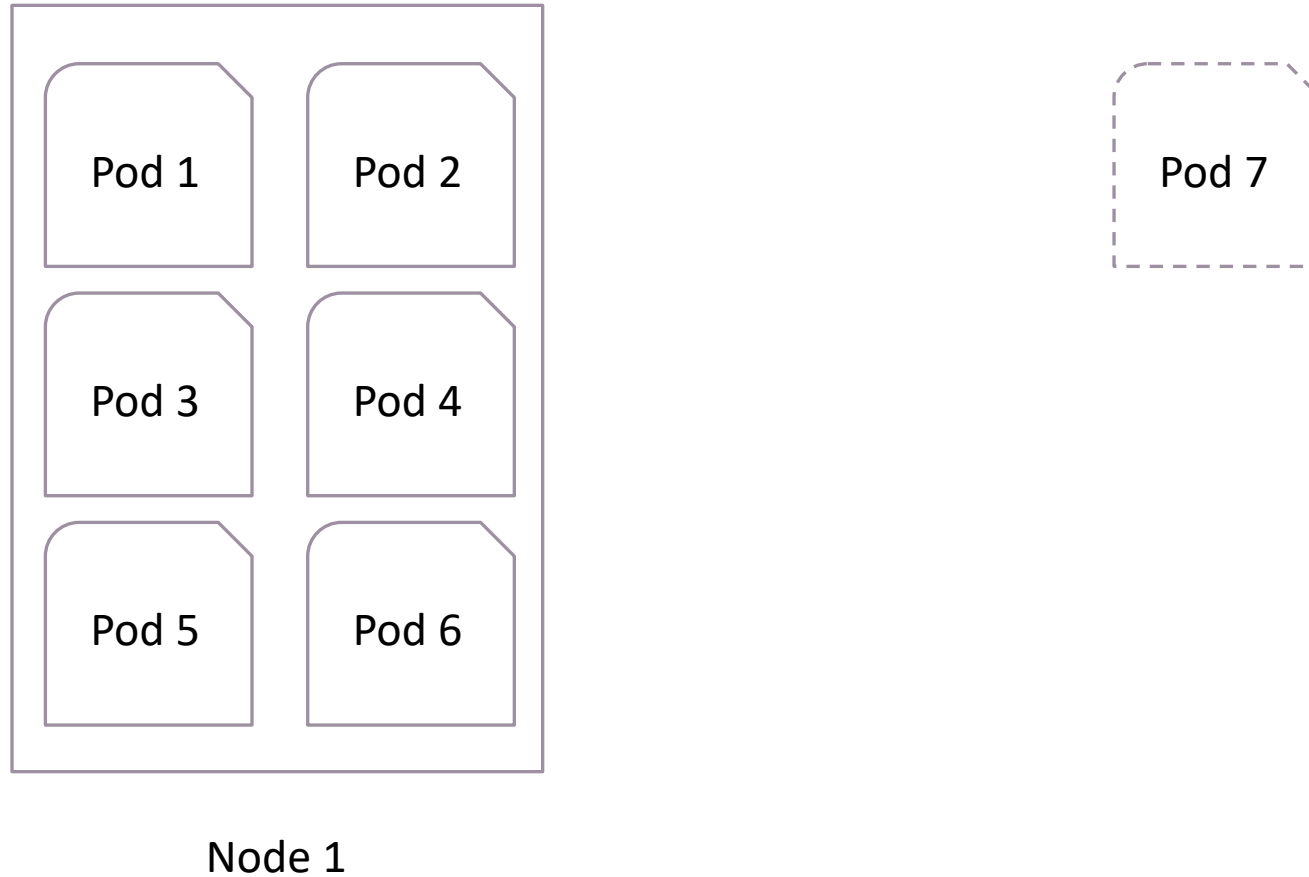
# Как работает Cluster Autoscaler: Upscale



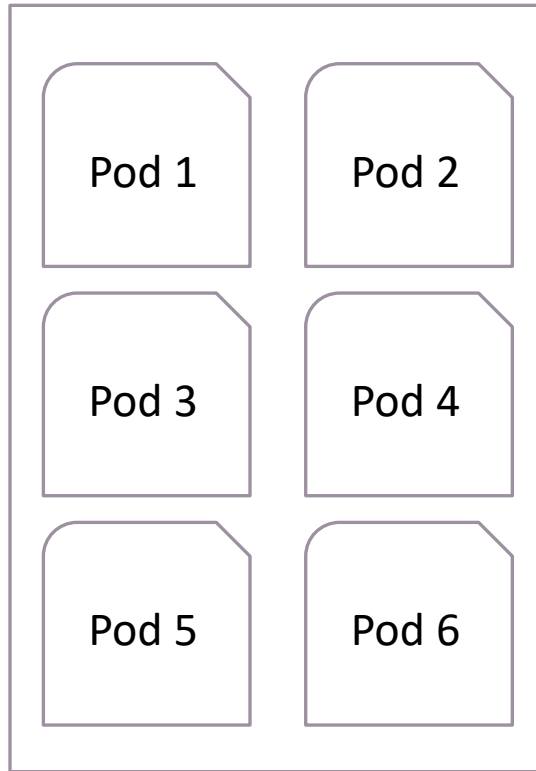
Node 1



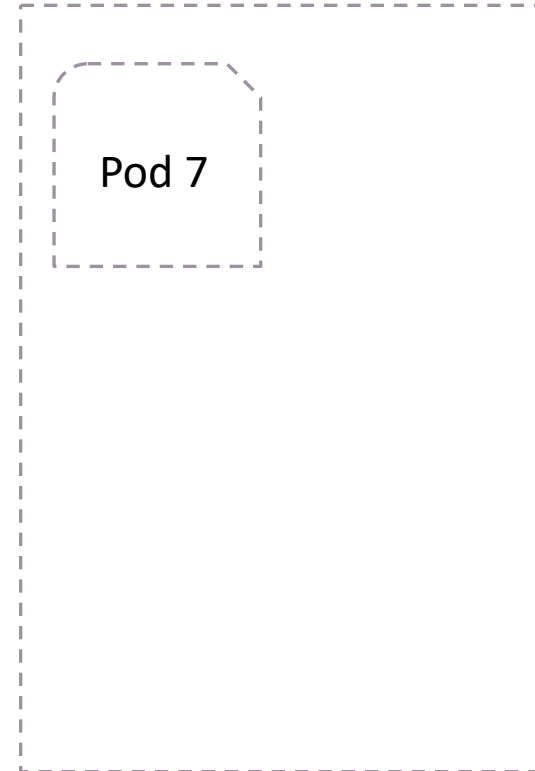
# Как работает Cluster Autoscaler: Upscale



# Как работает Cluster Autoscaler: Upscale

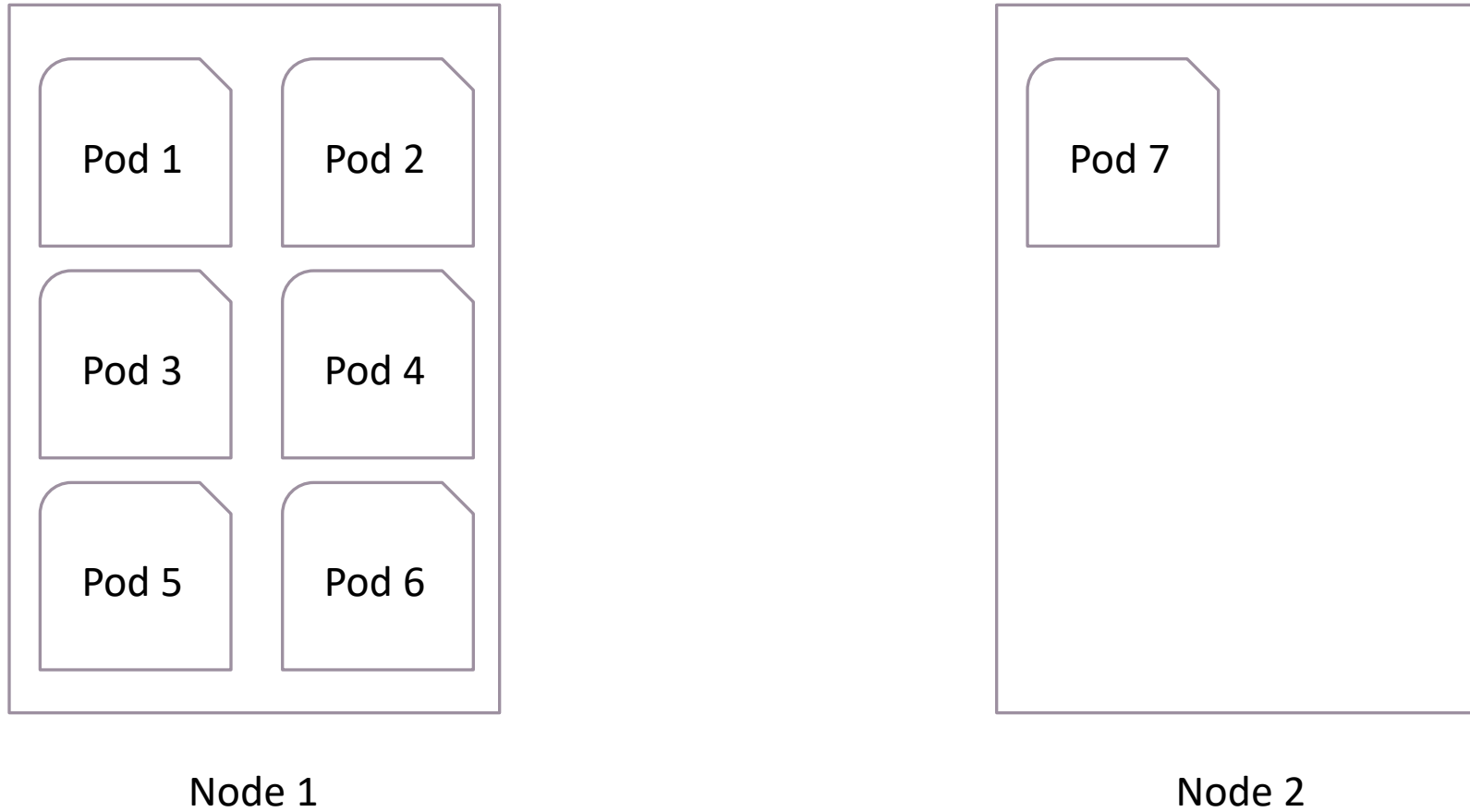


Node 1

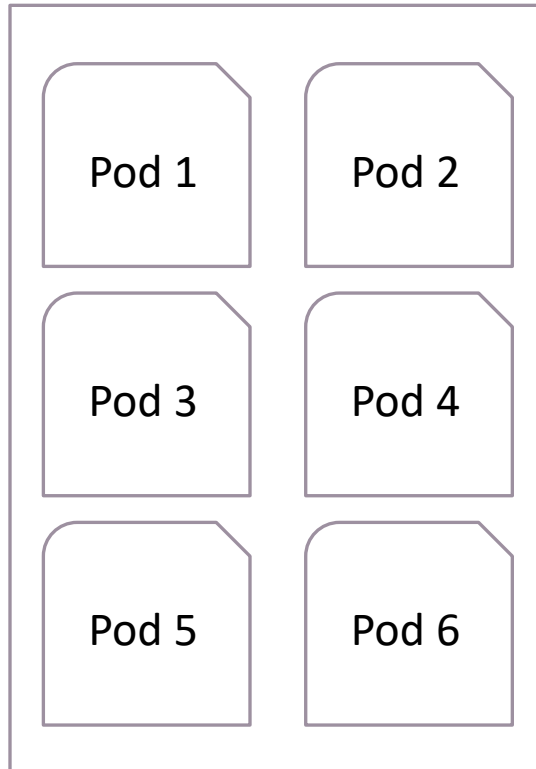


Node 2

# Как работает Cluster Autoscaler: Upscale



# Как работает Cluster Autoscaler: Downscale

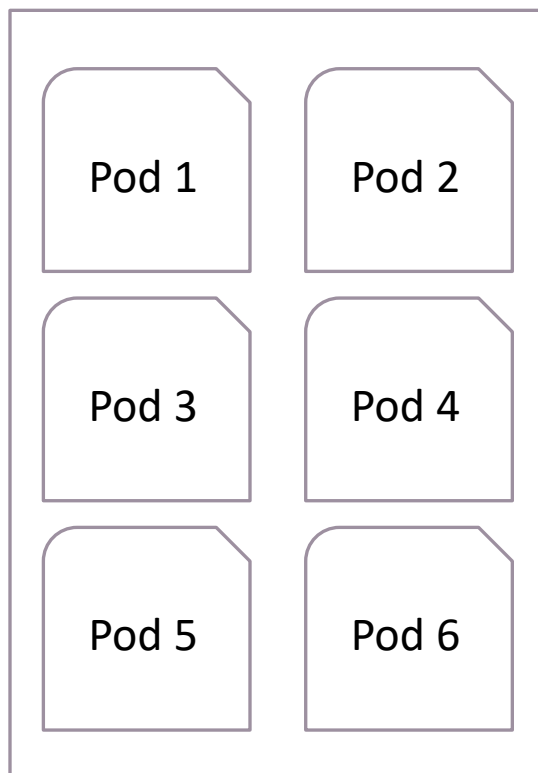


Node 1



Node 2

# Как работает Cluster Autoscaler: Downscale



Node 1

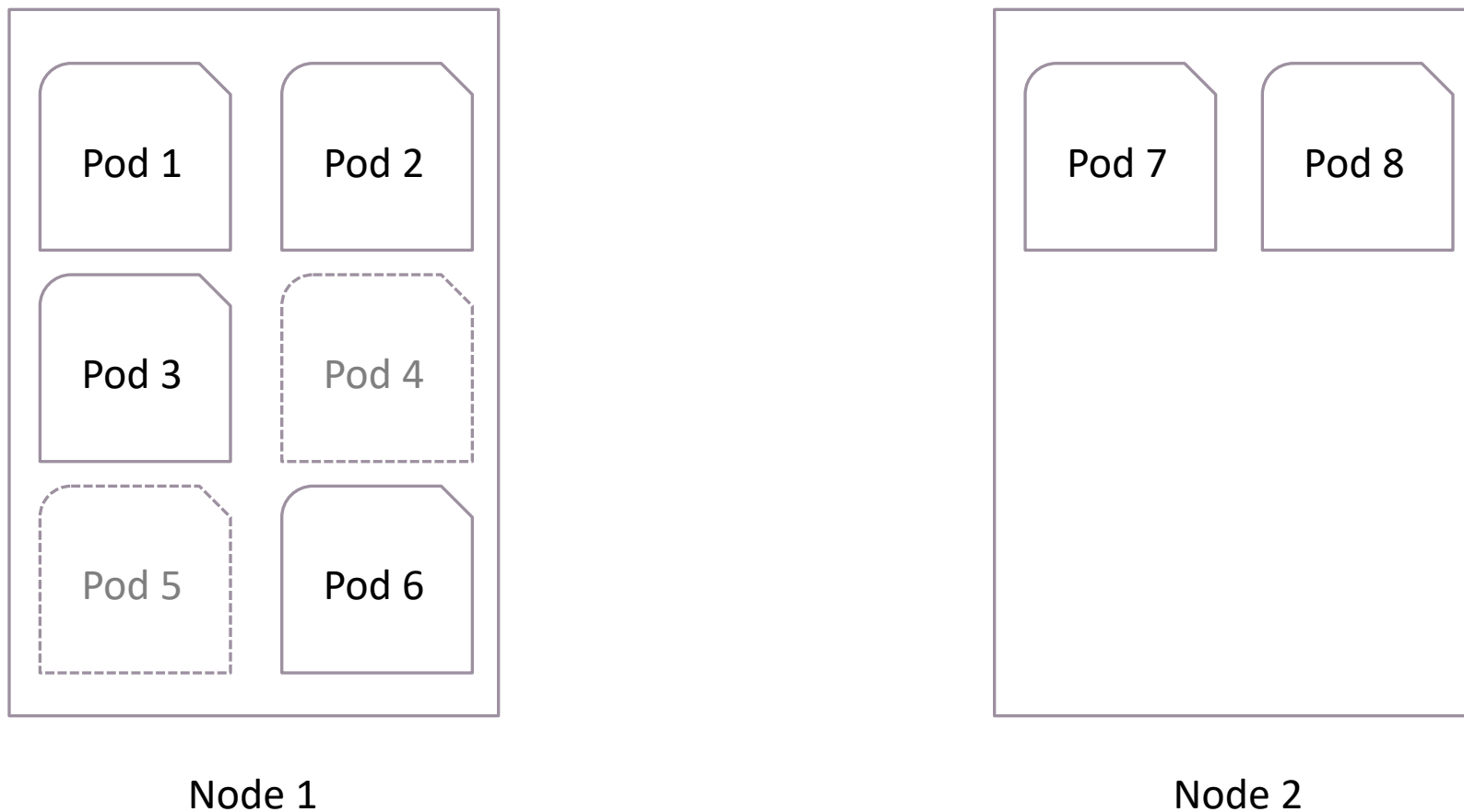


Node 2

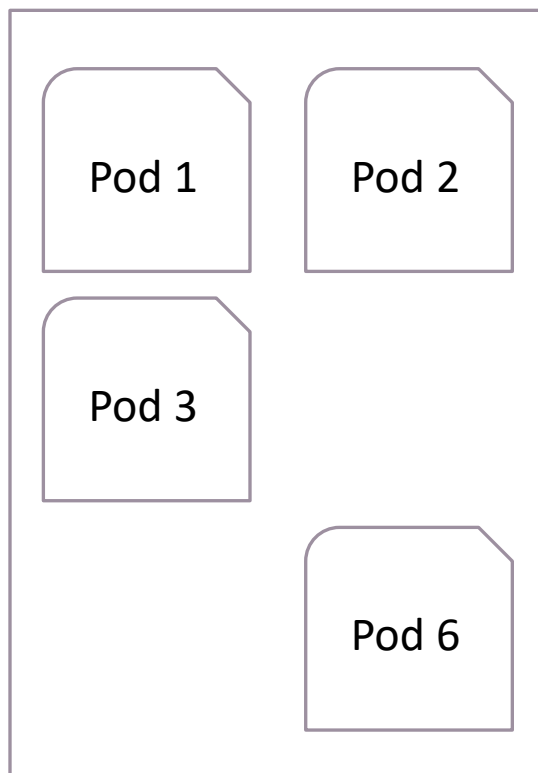


Но ведь так не бывает

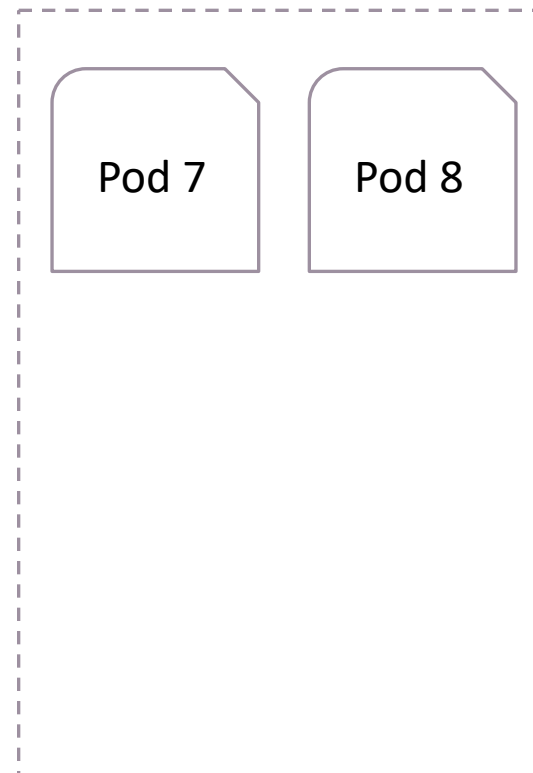
# Как работает Cluster Autoscaler: Downscale



# Как работает Cluster Autoscaler: Downscale



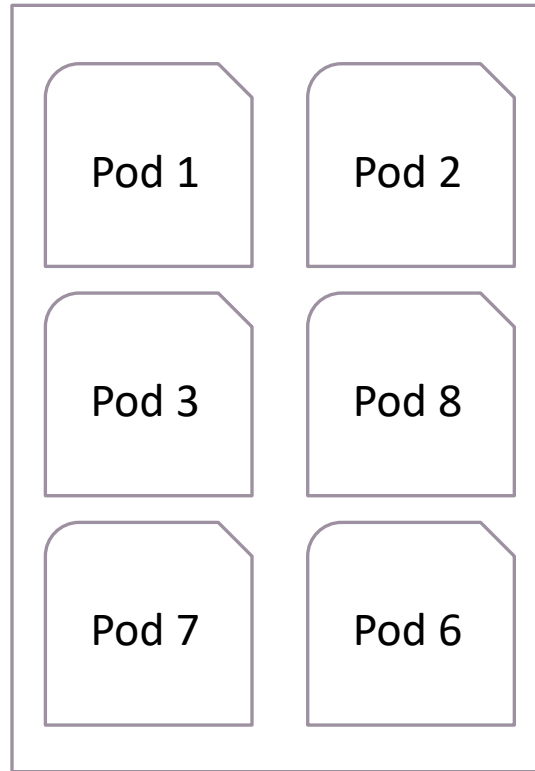
Node 1



Node 2



# Как работает Cluster Autoscaler: Downscale



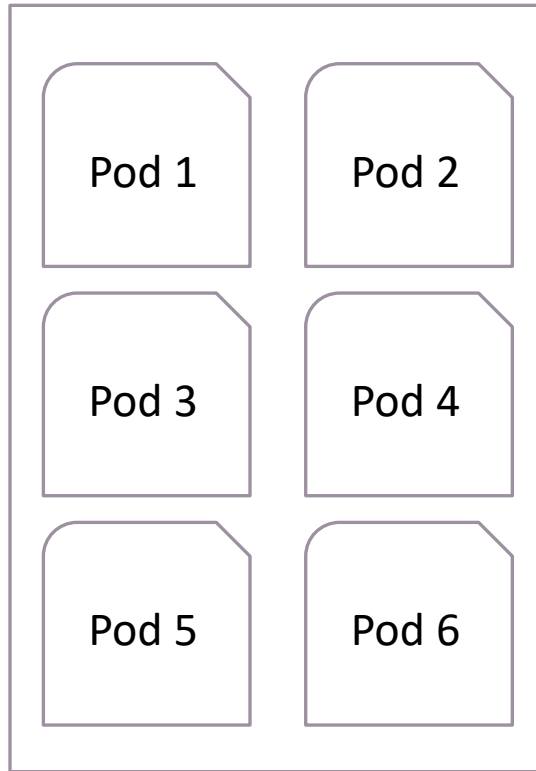
Node 1



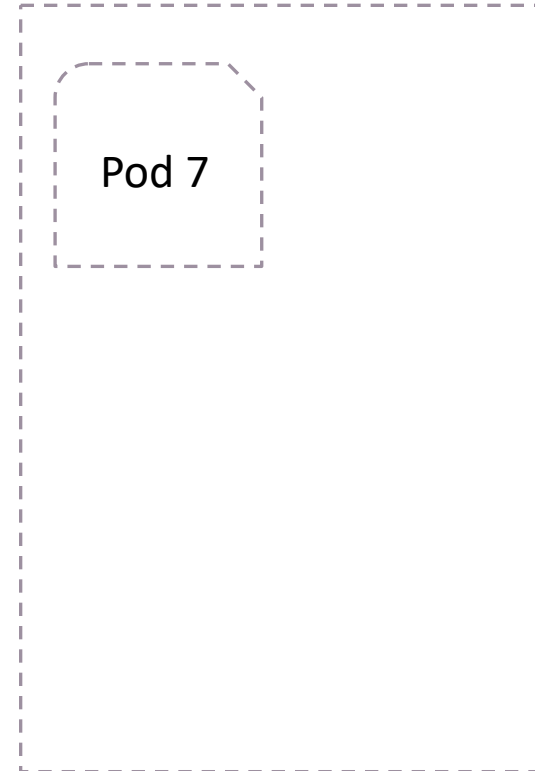
Node 2

\* Учитывается PodDisruptionBudget и GracefulTermination

# Cluster Autoscaler: Overprovision



Node 1

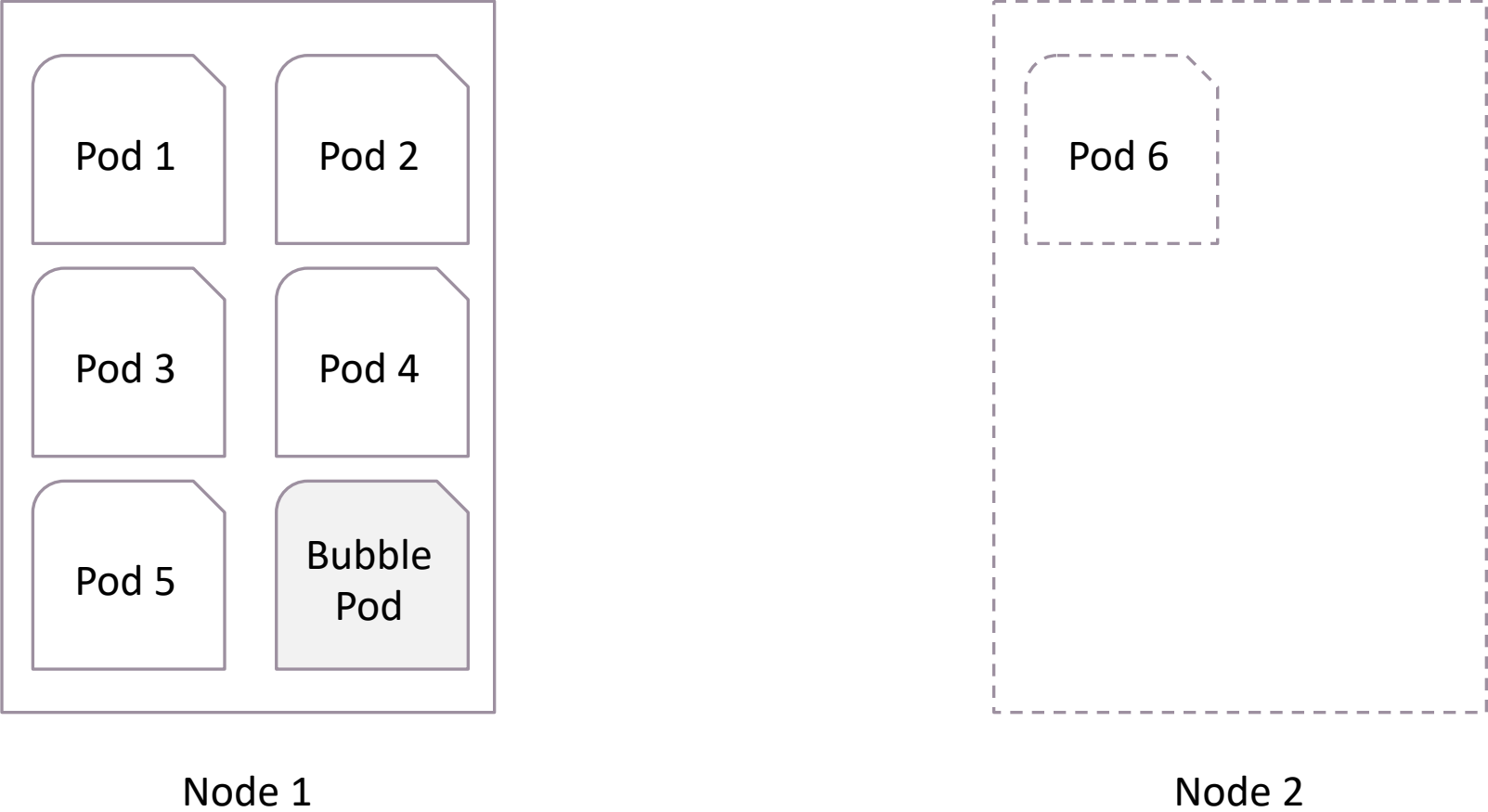


Node 2

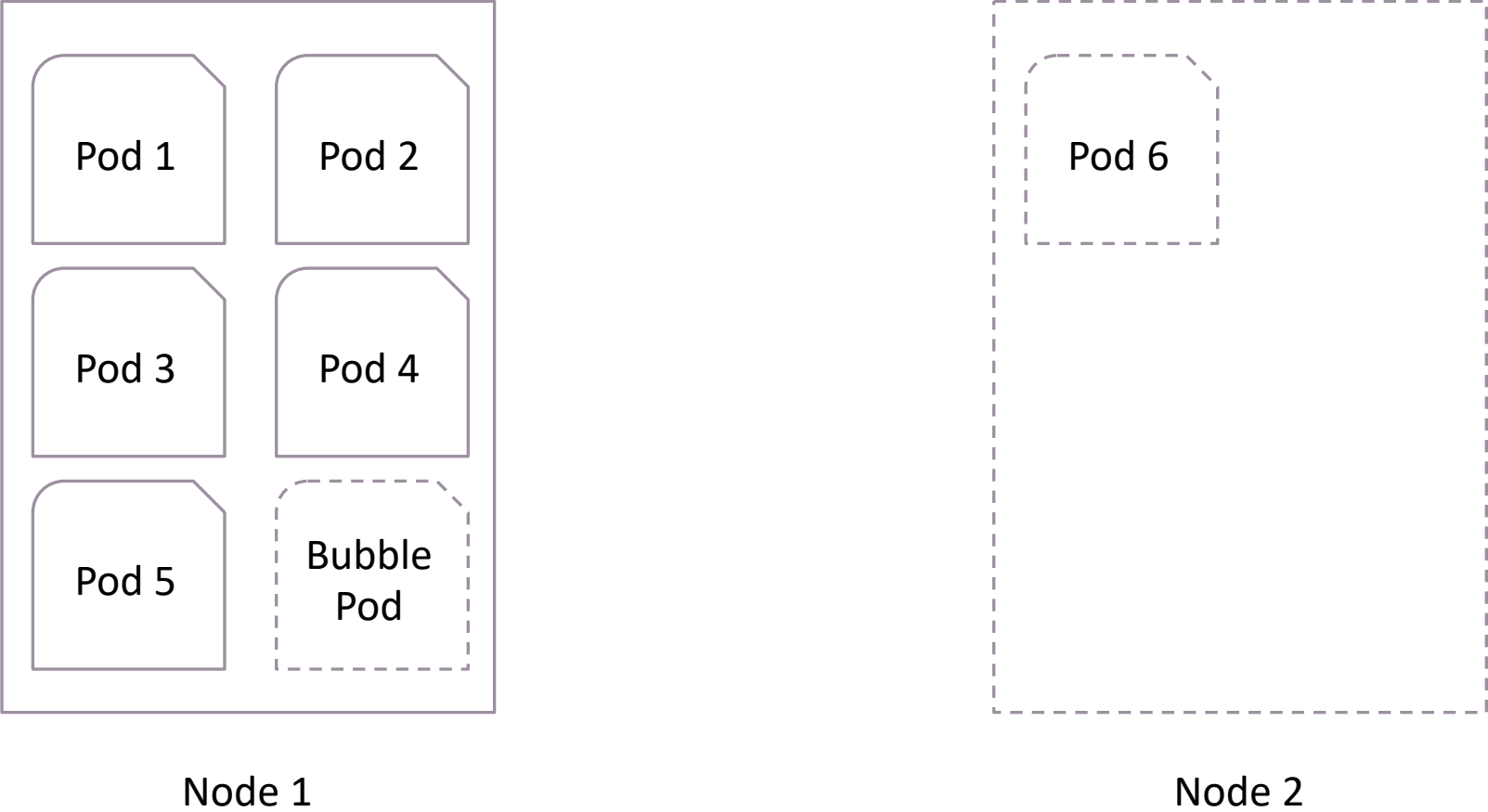


3 минуты

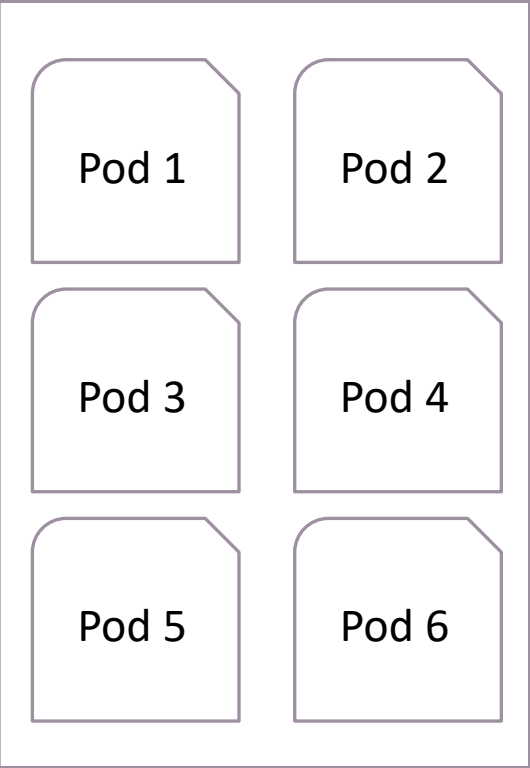
# Cluster Autoscaler: Overprovision



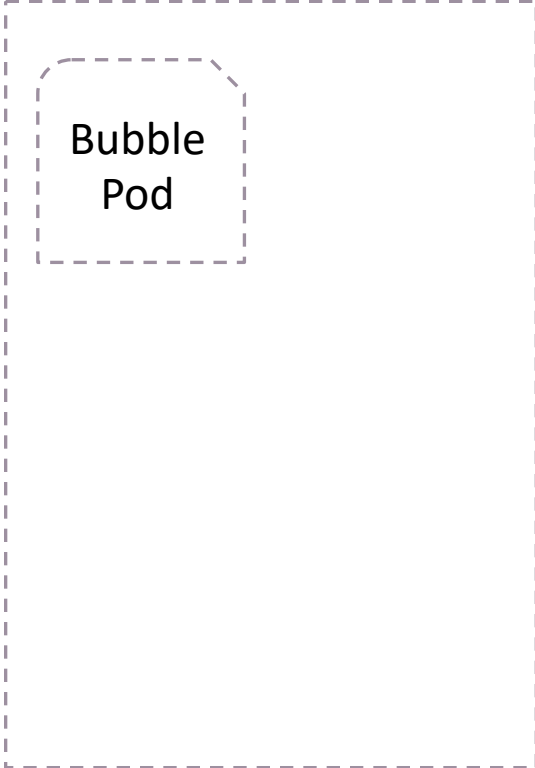
# Cluster Autoscaler: Overprovision



# Cluster Autoscaler: Overprovision

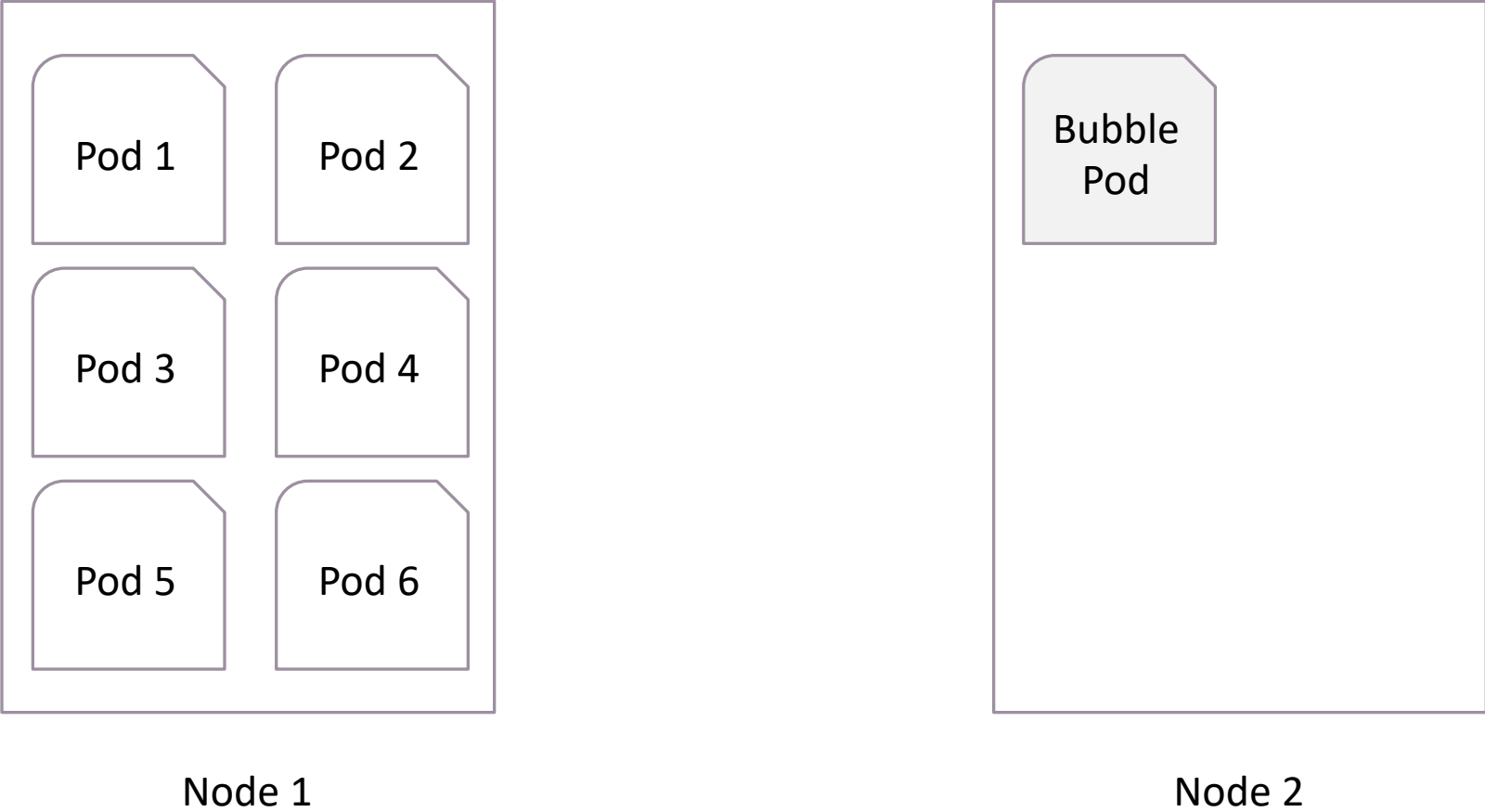


Node 1



Node 2

# Cluster Autoscaler: Overprovision



# Cluster Autoscaler: Overprovisioning

```
apiVersion: scheduling.k8s.io/v1beta1
kind: PriorityClass
metadata:
  name: overprovisioning
value: -1
globalDefault: false
description: "Priority class used by overprovisioning."
```

```
apiVersion: apps/v1
kind: Deployment
metadata:
  name: overprovisioning
  namespace: default
spec:
  replicas: 1
  selector:
    matchLabels:
      run: overprovisioning
  template:
    metadata:
      labels:
        run: overprovisioning
    spec:
      priorityClassName: overprovisioning
      containers:
      - name: reserve-resources
        image: k8s.gcr.io/pause
        resources:
          requests:
            cpu: "200m"
```

# Scale up from zero nodes

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "autoscaling:DescribeAutoScalingGroups",
        "autoscaling:DescribeAutoScalingInstances",
        "autoscaling:DescribeLaunchConfigurations",
        "autoscaling:DescribeTags",
        "autoscaling:SetDesiredCapacity",
        "autoscaling:TerminateInstanceInAutoScalingGroup"
      ],
      "Resource": "*"
    }
  ]
}
```

ec2:DescribeLaunchTemplateVersions

Как autoscaler узнает о типе инстанса:

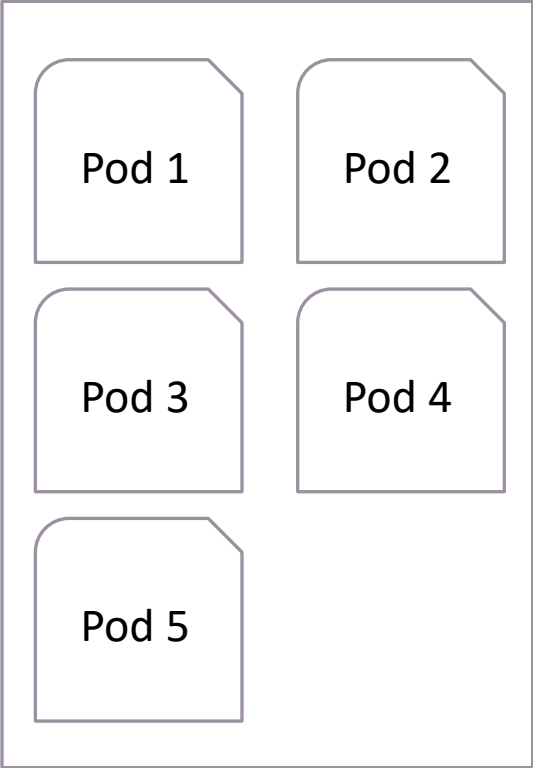
- 1) Launch Configuration
- 2) Launch Template
- 3) Смешанные типы



One more thing...



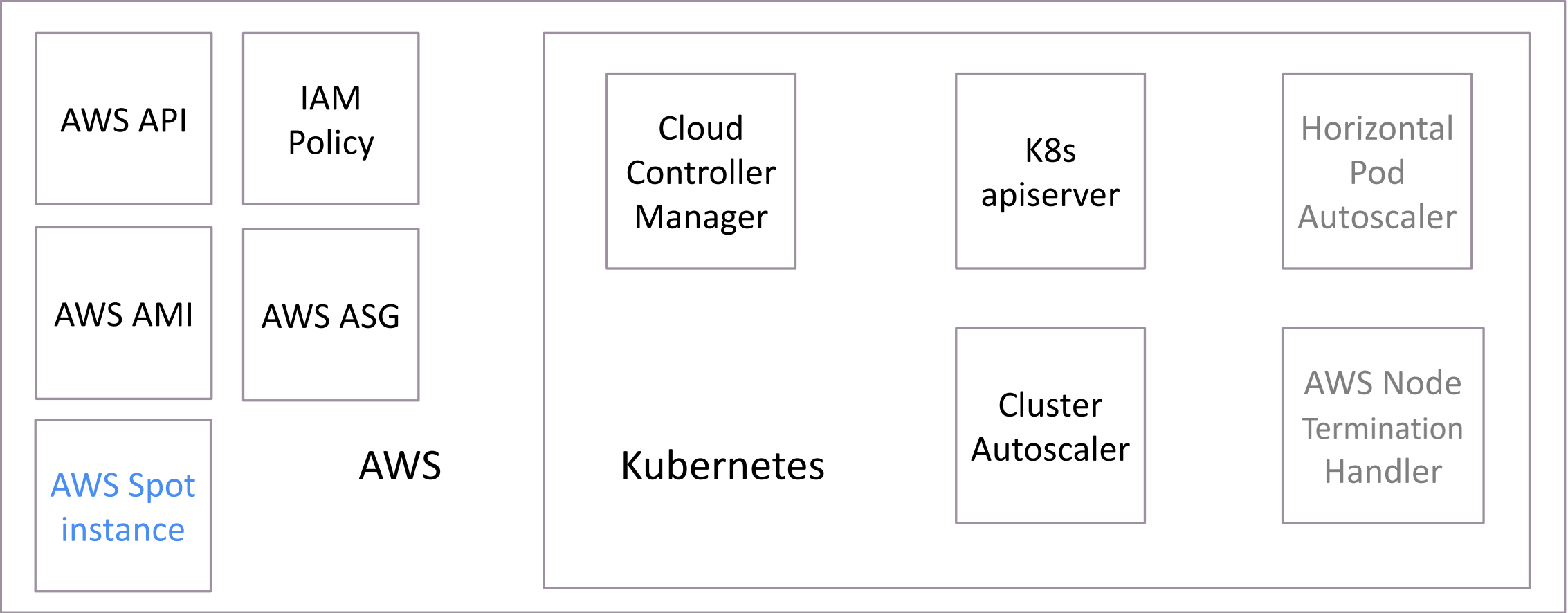
# Cluster Autoscaler



Node 1

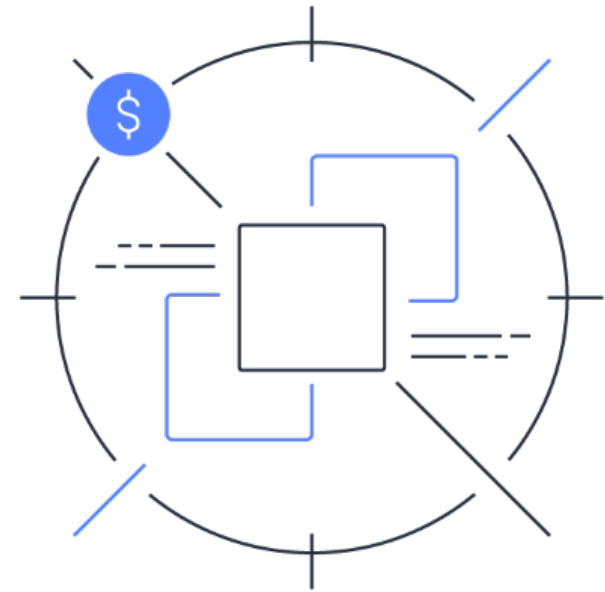


# Архитектура решения



# AWS – Spot instances

- До 80% дешевле, чем On-Demand
- Спроектирован для нагрузки, ГОТОВОЙ К ОСТАНОВКАМ



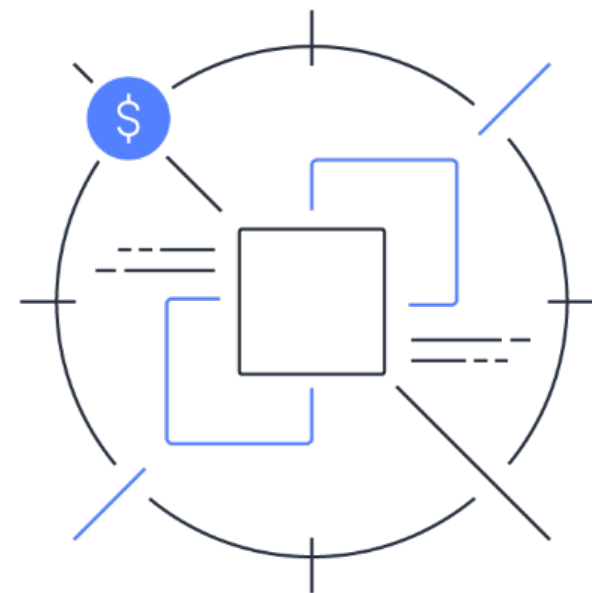
# AWS – Spot instances

В среднем за последнюю  
неделю t3a.medium

**0.017\$** в час

Обычная цена:

**0.037\$** в час

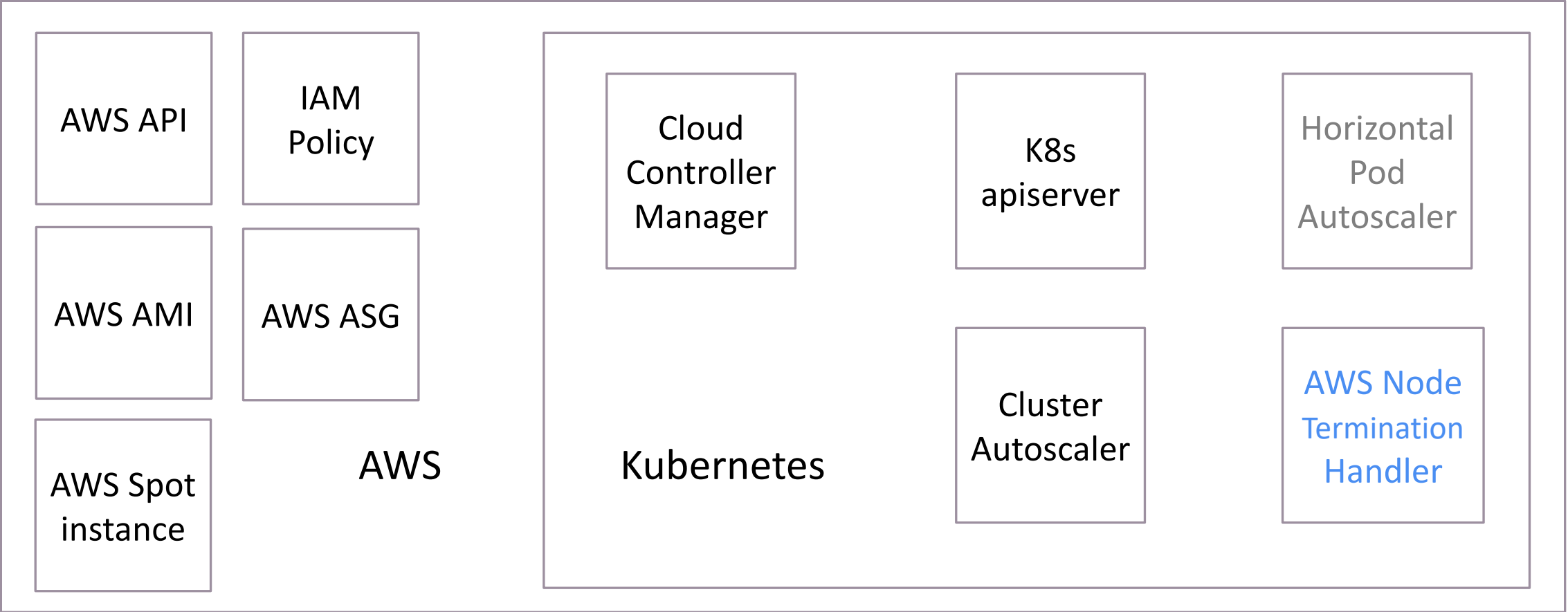


\*Данные за неделю 23-29 ноября 2020, EU Central

# AWS ASG Spot instances proportion

- On-Demand base: 3
- On-Demand percentage above base: 0%
- Spot percentage: 100%

# Архитектура решения

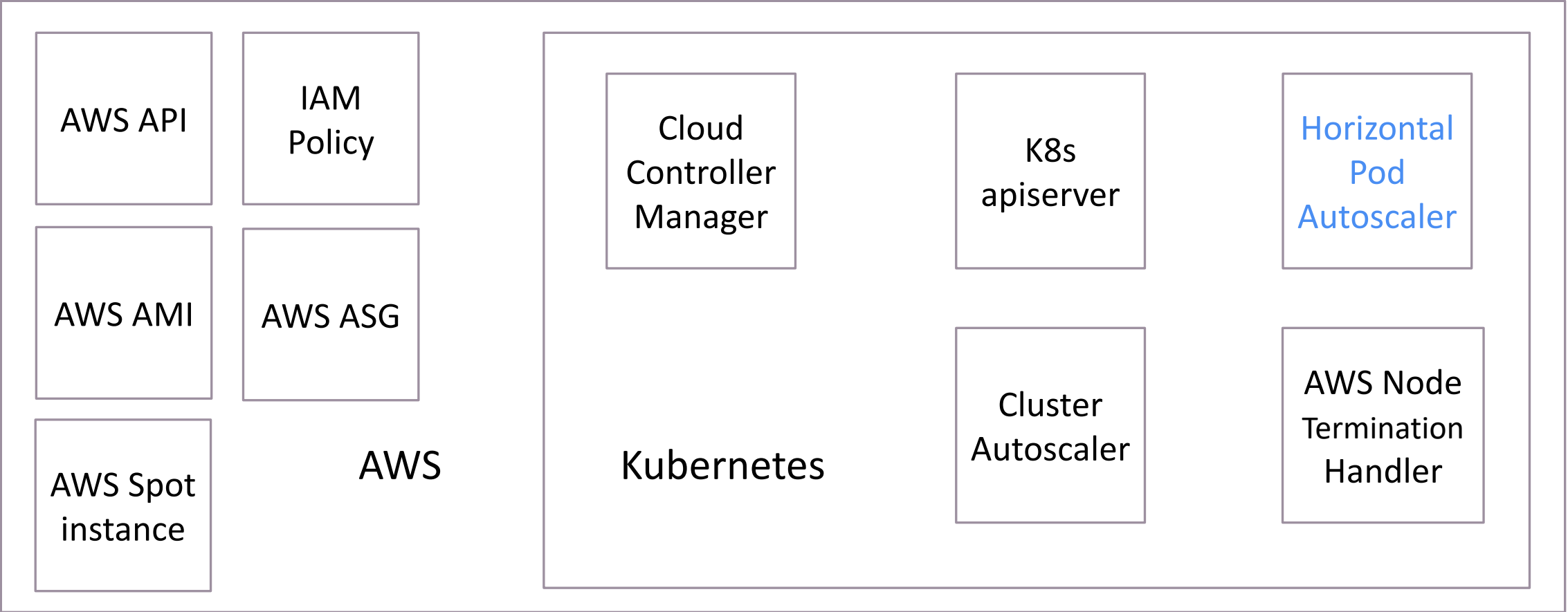


# AWS Node Termination Handler

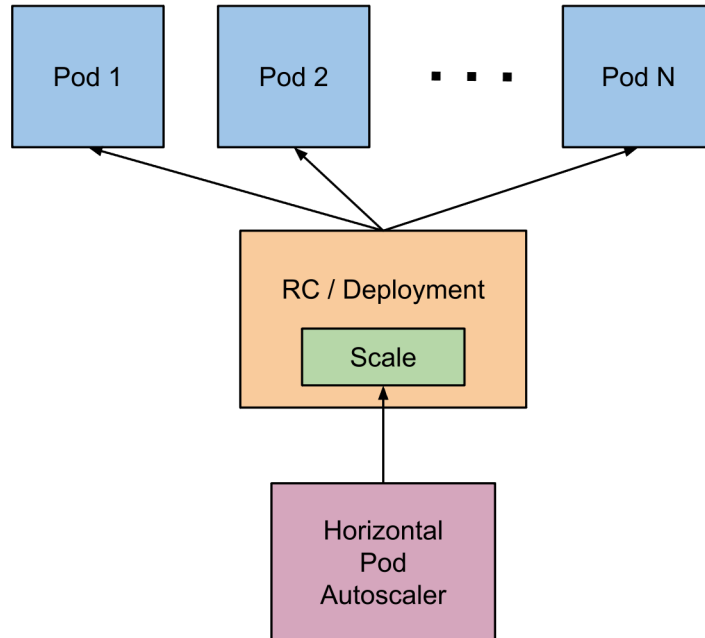
- AWS может отобрать Spot Instance
- Предупреждает за 2 минуты
- Выполняется drain k8s ноды и перераспределение нагрузки



# Архитектура решения



# У меня уже есть HPA – не помешает?



## Horizontal Pod Autoscaler

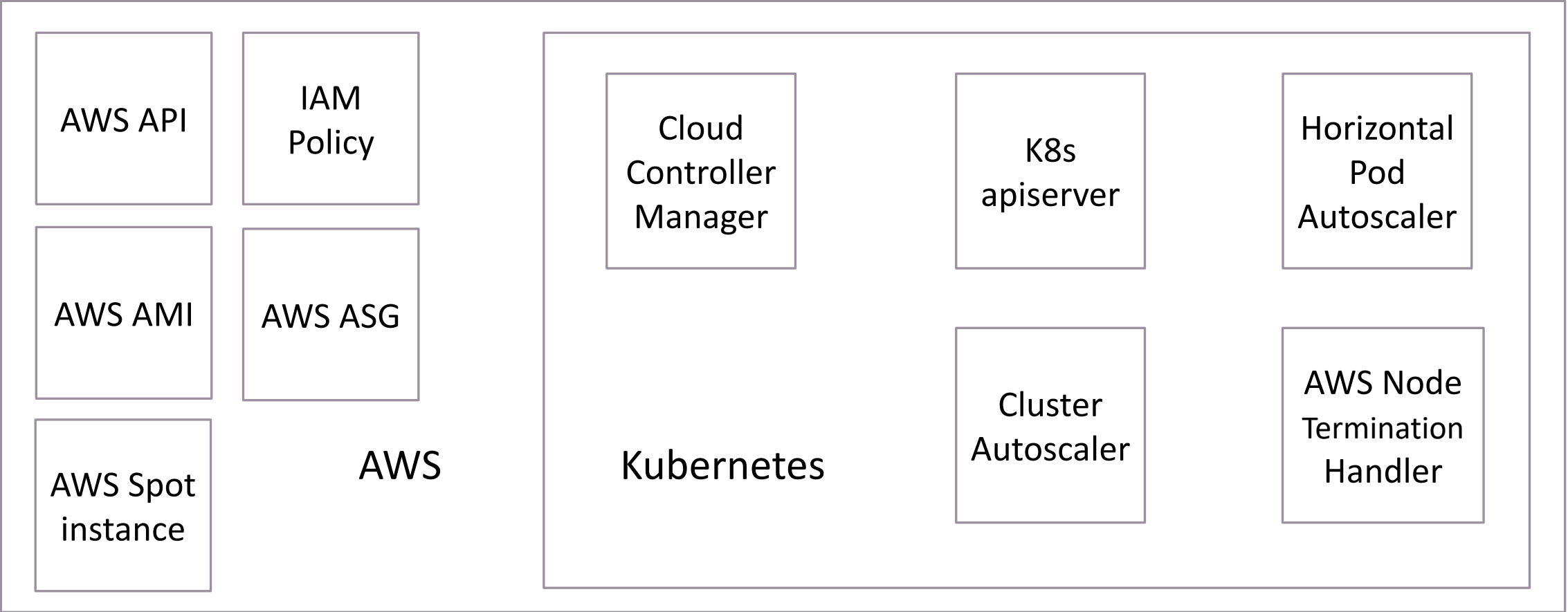
Инструмент, позволяющий  
увеличивать/уменьшать количество подов в  
replica set/deployment/statefulset



Доклад Ивана Глушкова

<https://www.youtube.com/watch?v=lpJgL-VRyaM>

# Архитектура решения



# Доставка: helm install



Cluster Autoscaler

<https://github.com/kubernetes/autoscaler/tree/master/charts/cluster-autoscaler-chart>



AWS Node Termination Handler

<https://github.com/aws/aws-node-termination-handler/tree/main/config/helm/aws-node-termination-handler>



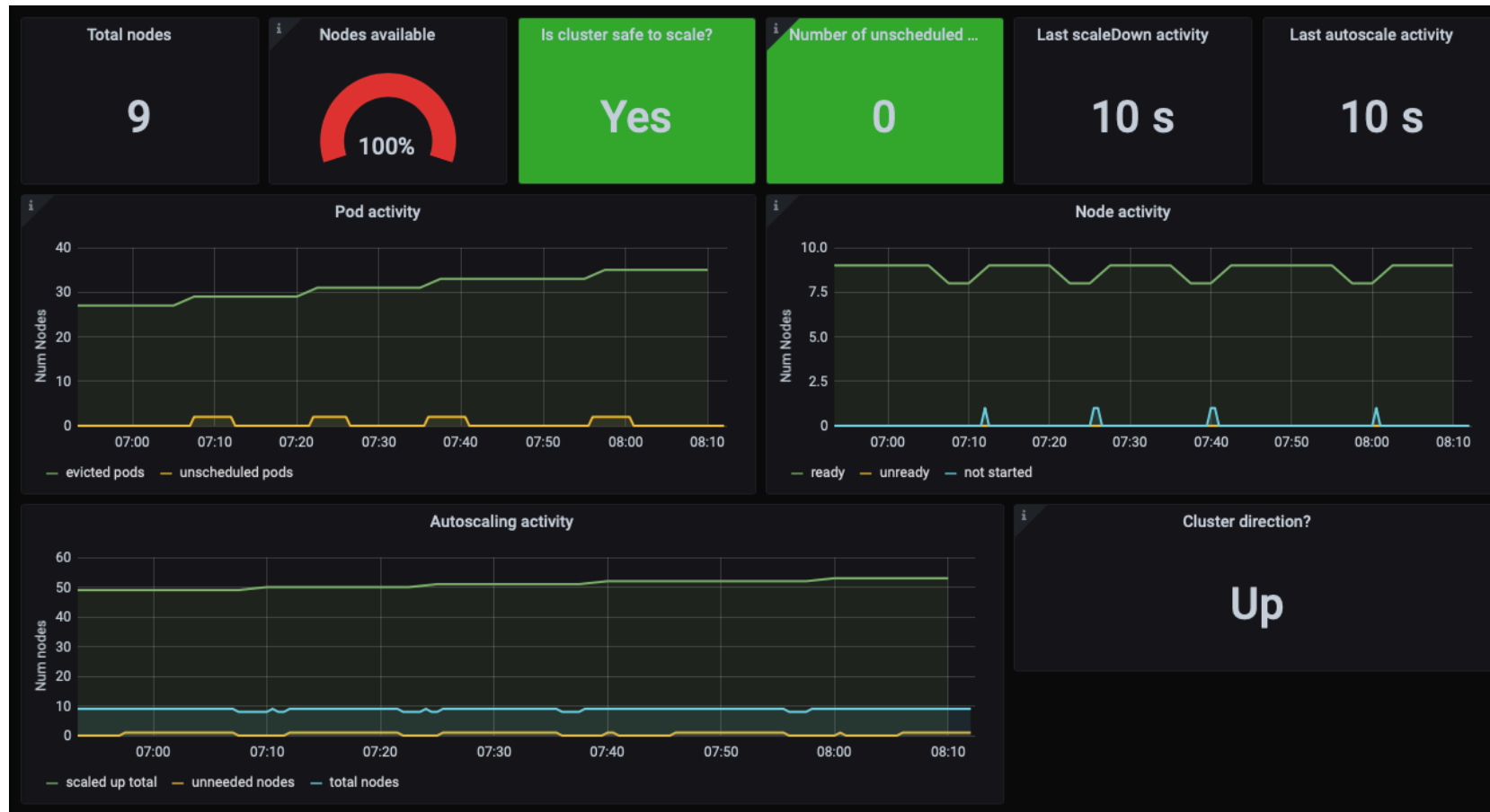
Horizontal pod autoscaler

<https://github.com/banzaicloud/hpa-operator/tree/master/deploy/charts/hpa-operator>

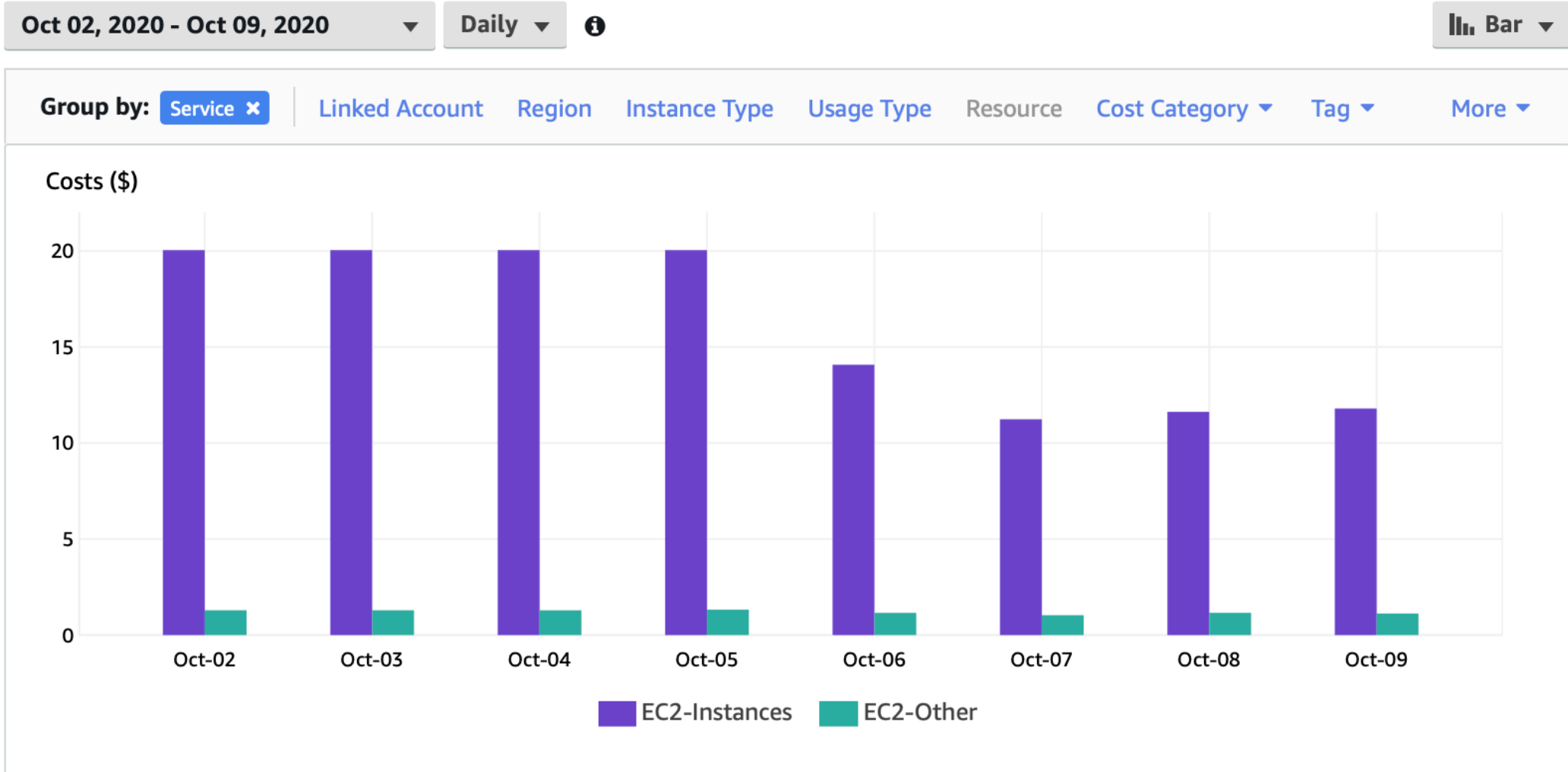


# Мониторинг Cluster Autoscaler

Сервис отдает метрики по эндпоинту /metrics



# Выгоды от внедрения решения



# Что в итоге?

- Знаете что умеет Cluster autoscaler и можете внедрить его у себя
- Можете повторить достигнутую нами экономию в 40% на Spot'ах

# Вопросы?

Viktor Erpylev

TechLead

Email:

[verpylev@plesk.com](mailto:verpylev@plesk.com)

Telegram:

velmoga

plesk

