

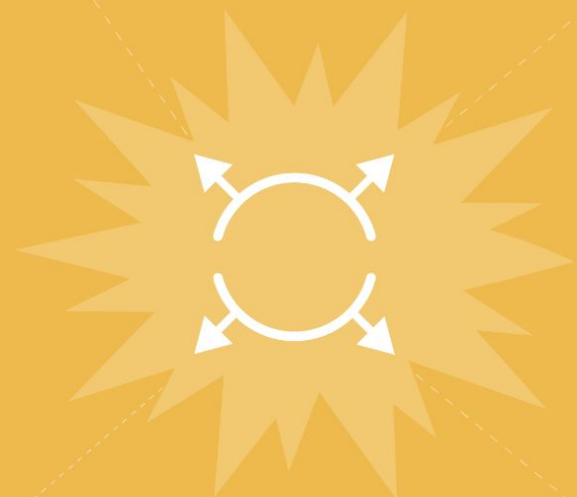
# PreScaler Operator

<https://github.com/ContainerSolutions/pre-scaling-operator>

30.06.2021



thomas.richter@container-solutions.com  
container-solutions.com



[PRESCALER OPERATOR]



**Thomas Richter**

***Cloud Native Engineer @ Container Solutions***



## Scaling for peak traffic can be difficult



Black Friday 2020, Worldwide

Indexed Daily Sales, compared to average in October 1-28 2020  
Same set of Retailers with stable sales tracking during the period in 2019 and 2020

Source: [Criteo](https://criteo.com)



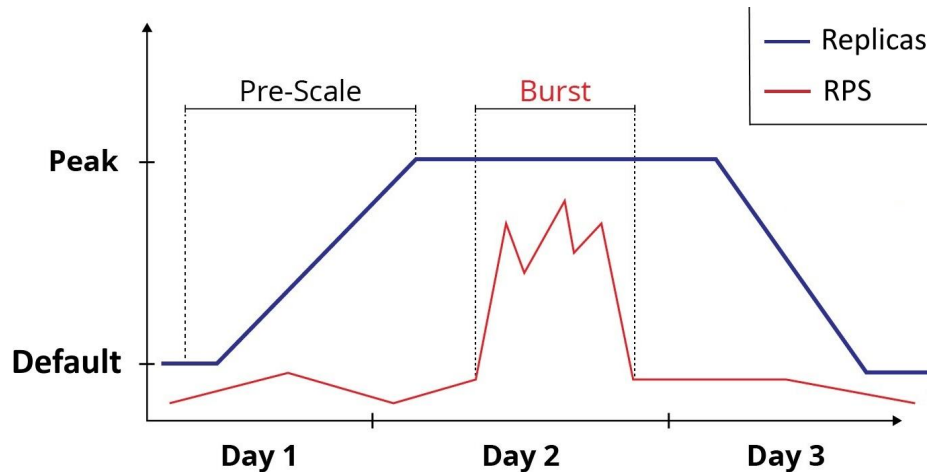
- Scaling rapidly to large numbers can be risky and unpredictable
- Different workloads have different scaling needs
- The HPA cannot cope with big bursts of traffic reliably





## The purpose of the operator

- Scale applications before receiving predictable load, without needing to change replicas

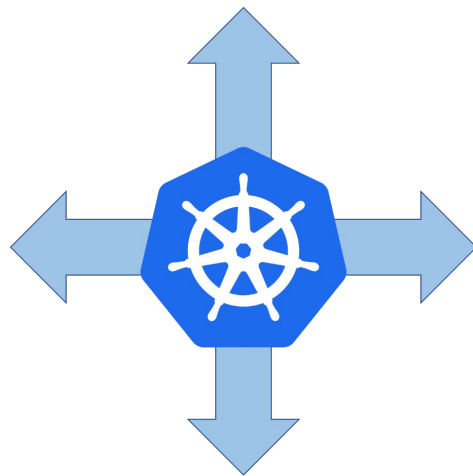


- The Pre-Scaler operator takes **ownership** of the `spec.replicas` field

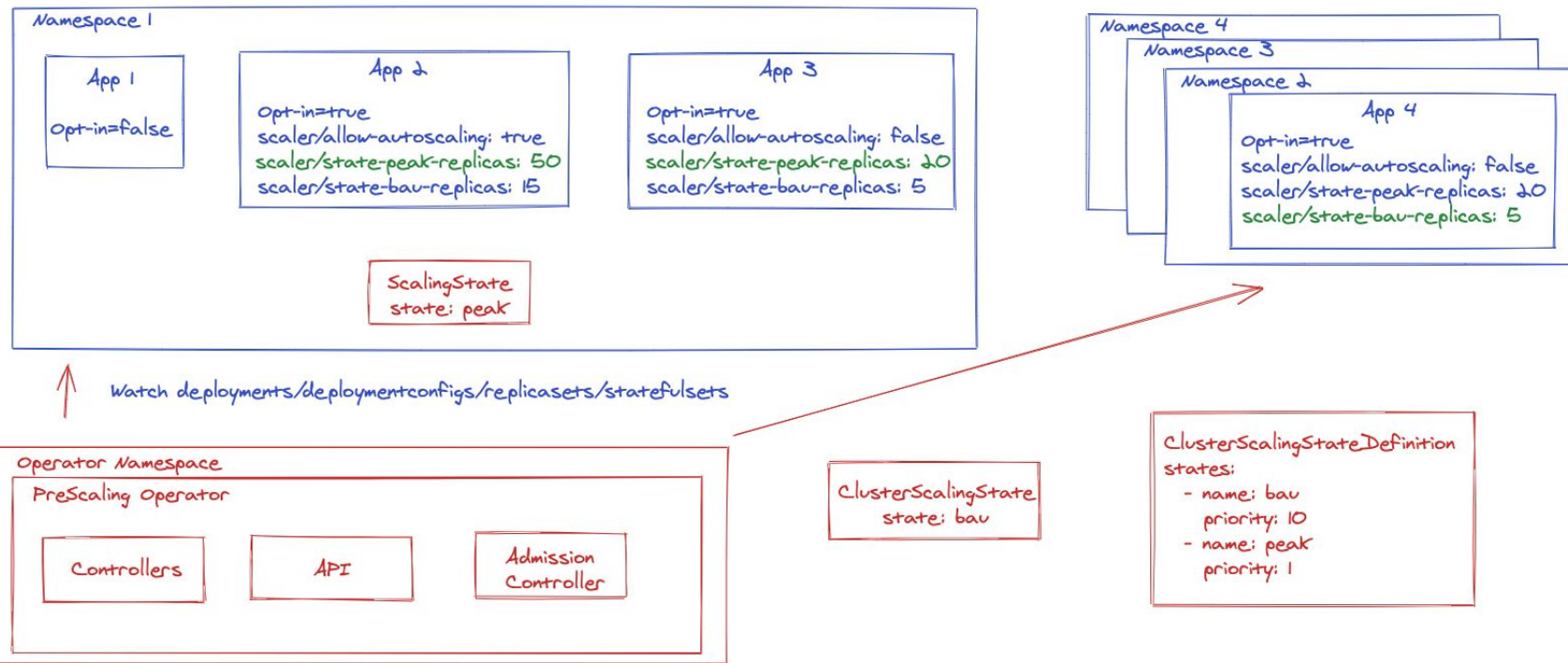


## How it works

- Support of 3 different scaling use cases:
  - No scaling allowed
  - PreScale to user defined number of replicas
  - PreScale and allow autoscaling upwards
- Kubernetes and Openshift native
- Ability to scale workloads in a single namespace or even whole cluster through an easy switch



## Cluster



ClusterScalingStateDefinition: List of possible states for namespace/cluster

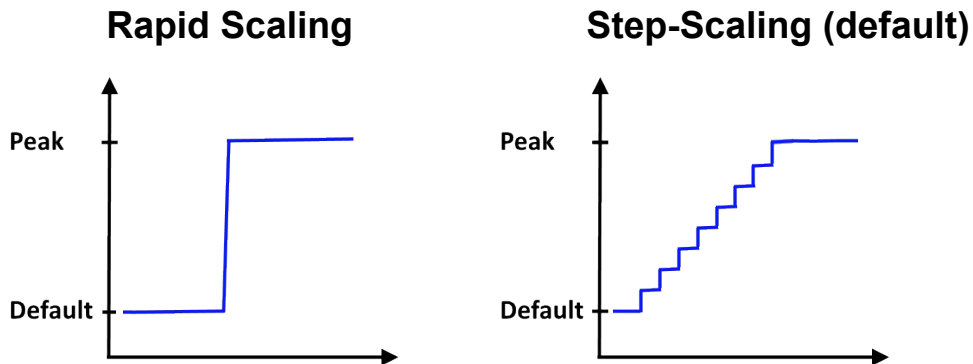
ClusterScalingState: Desired cluster state

ScalingState: Desired namespace state



# Feature overview

- Rapid Scaling or Step Scaling



- Takes Resource Quotas into account
  - In case of violation: The namespace won't be scaled



## Feature overview part two

- State prioritization (Namespace states vs cluster-wide states)
  - Scale namespaces individually or the whole cluster
- DryRun mode
  - See what **would** happen
- Limit number of Namespaces to be scaled at the same time



## Future features...

- Support for more resources:
  - ReplicaSet
  - StatefulSet
- Support to ignore certain namespaces
- Offer support to make more detailed scaling decisions
  - Based on certain namespace labels
- A simple UI
- Metrics
- Scheduled scaling
- And more..



## Where are we currently

- The first pre-release is already out!
- The main aim is to get real user feedback
- We are looking to grow the community and its user base
- We are targeting summer 2021 for production readiness

**Feel free to contribute:** <https://github.com/ContainerSolutions/pre-scaling-operator>



**Let's play!**  
A demo