pepperdata

Pepperdata Capacity Optimizer for Microservices on Kubernetes

Achieve Immediate Utilization Improvements and Reduced Costs

The Challenge: Resource Utilization in Long-Running Services

The configuration of resource requests (CPU and memory) for any workload directly influences its performance, stability, and associated costs. DevOps teams often add large buffers to ensure their workloads run to completion, leading to overallocated resources and wasteful spending—and this applies to services, too.

The conventional practice of manual configuration requires constant, time-consuming monitoring and adjustments. Since resource requirements fluctuate constantly, these manual settings quickly become outdated, leading to wasted resources or performance issues that are difficult, if not impossible, to manage manually—especially at scale.

The Solution: Dynamic Resource Optimization

Pepperdata Capacity Optimizer is a dynamic resource optimization solution that automatically aligns the resource allocations of service pods with actual resource usage to run more pods per node, resulting in increased utilization and reduced operating costs.

Aggressive bin packing of pods can boost utilization, but this strategy carries an increased risk of errors due to potential sudden spikes. Therefore, reducing pod requests without careful consideration is hazardous. Capacity Optimizer avoids this by monitoring individual node utilization to ensure that a node is never under resource pressure due to resource adjustments. Capacity Optimizer also enhances cloud autoscaling efficiency by ensuring that new nodes are provisioned only when the existing nodes are fully utilized.

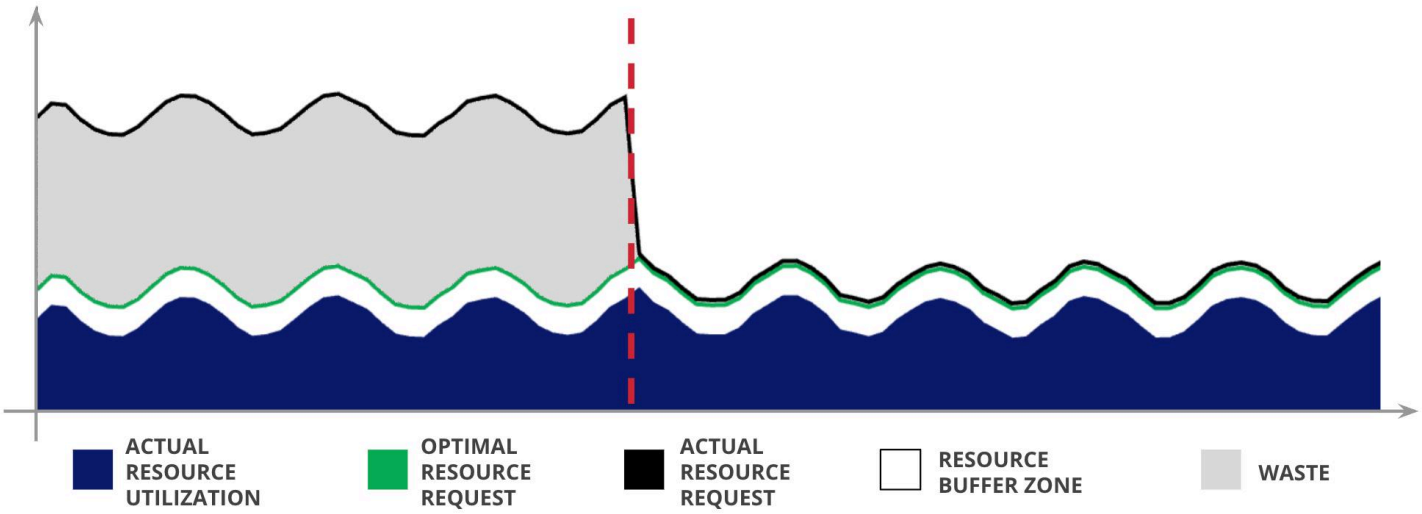

Figure 1: Before Capacity Optimizer is enabled, resource requests are not aligned with resource utilization. Once enabled, Capacity Optimizer aligns the resource allocations with actual resource usage to provide more pods per node, increased utilization, and reduced costs.

How Capacity Optimizer Enhances the Autoscaler

Capacity Optimizer operates at the service level and functions like a Vertical Pod Autoscaler (VPA), but unlike VPA, it works in real time. Note: To avoid conflicts, any existing VPA should either be disabled or set to recommendation-only mode.

- Capacity Optimizer automatically reduces the resource requests for a service's existing pods and ensures that new pods are launched with reduced resource requests. This optimization is achieved without the need for any service restarts.
- With Capacity Optimizer actively reducing pod resource requests based on actual usage, more pods can be scheduled onto existing nodes—maximizing node utilization.
- Because pods are launched with reduced resource requests, the autoscaling mechanism is inherently optimized, in contrast to solutions that must reduce pod requests after launch.

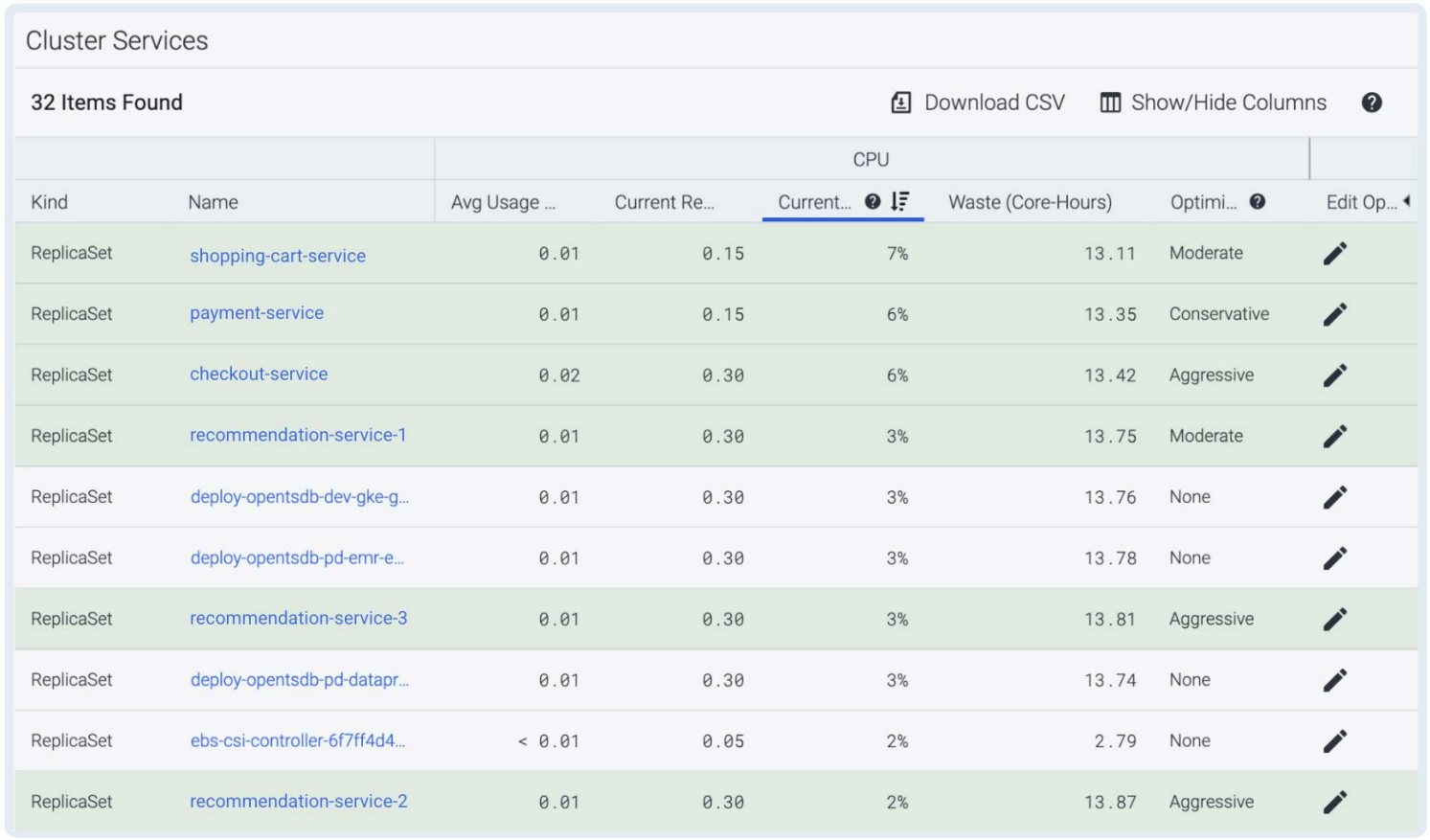

Figure 2: Via Pepperdata's dashboard, you can choose the desired optimization level for individual services (conservative, moderate, aggressive, or none).

FAQs

Which versions of Kubernetes does Pepperdata recommend?
Peppedata recommends Kubernetes version 1.33 or higher, which includes support for in-place pod resizing. In-place pod resizing can result in significantly improved cost savings. However, Pepperdata supports earlier versions of Kubernetes as well.

Does Capacity Optimizer use in-place pod resizing?
Yes, along with pod mutation before the pods are launched. In-place pod resizing requires Kubernetes version 1.33 or higher.

What optimization levels are available?
Depending on the desired balance between cost reduction and the potential impact on a service's performance, you can select a conservative, moderate, or aggressive optimization level.

Does Capacity Optimizer restart a service's running pods when updating the optimization level?
No.

If the optimization level for a long-running service is updated, how long does it take before the change is applied?
Approximately five minutes.

If the optimization level for a long-running service is updated, is this change only applied once?
No. The update is reapplied approximately every five minutes to align allocations with actual resource utilization.

Capacity Optimizer Benefits

Real-Time Reclamation of Resource Waste
Align resource requests close to actual hardware usage to reduce waste and lower cloud bills, without impacting response time.

Increased Utilization Without Disruption
Pack more pods on existing physical nodes by working seamlessly with HPA and without any pod restarts.

Enhanced Autoscaling Efficiency
Launch new nodes only when existing nodes are optimally utilized.

Supported Technologies

- Amazon EKS, GKE, Microsoft Azure
- Static clusters or autoscaled clusters
- Cluster Autoscaler, Karpenter, GKE NAP
- Horizontal Pod Autoscaler (HPA)
- Kubernetes Event Driven Autoscaler (KEDA)
- GitOps tools that control the service lifecycle
- Supported Kubernetes controllers: Deployments, ReplicaSet

About Pepperdata

Pepperdata delivers dynamic resource optimization for Kubernetes workloads and AI infrastructure—on premises, in the cloud, and for GPUs. Since 2012 Pepperdata has helped companies ranging from startups and mid-sized ISVs to top enterprises such as Citibank, Autodesk, Magnite, Royal Bank of Canada, and members of the Fortune Five save over $250 million. Learn more at pepperdata.com.

aws partner network
Select Technology Partner

AICPA SOC aicpa.org/soc
Formerly SAS 70 Reports

100% ROI GUARANTEE

Pepperdata, Inc.
530 Lakeside Drive
Suite 170
Sunnyvale, CA 94085

Start a Free Trial
www.pepperdata.com

Send an Email
info@pepperdata.com