



eBook

The top 6 challenges to optimized Kubernetes infrastructure

How to eliminate complexity and streamline container management





Introduction

Kubernetes has become the preferred platform for managing containers across an organization's diverse cloud environments. However, many businesses struggle to realize the full return on their investment in Kubernetes due to the challenges of scaling this transformative technology. Engineering teams face significant obstacles in optimizing infrastructure efficiency and minimizing operational overhead at scale. Common issues such as waste, sprawl, and complexity often leave teams struggling to control costs while grappling with uncertainty about whether their infrastructure can adapt to the evolving demands of their applications.

Additionally, teams often lack a comprehensive understanding of cloud spending and resource utilization across different teams and cloud environments. Furthermore, they can also face limitations in skills, tooling, budget, and staff, making it difficult to effectively balance cost, availability, and performance across their organization's cloud-native infrastructure.

To eliminate complexity and streamline container management, organizations must focus on the continuous optimization of resource allocation to meet dynamic application demands effectively.

Spot by NetApp's Ocean unifies, streamlines, and automates the management of large-scale Kubernetes clusters across multiple clouds and hybrid environments. A serverless infrastructure engine for containers, Ocean automatically ensures each workload has its uniquely optimized mix of spot, reserved, and on-demand instances.

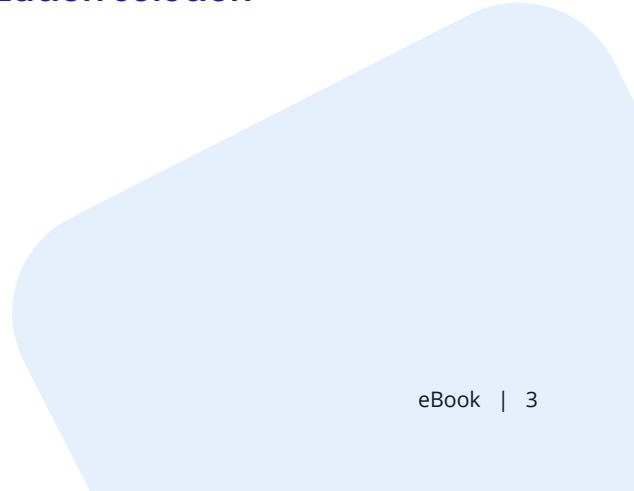
High-level benefits of Ocean

- Streamlines Kubernetes infrastructure management by offloading the complexity of managing resources from engineering teams
- Balances Kubernetes infrastructure performance, availability, and cost through continuous, automated optimization
- Delivers cost visibility, observability, and control via detailed insights into cloud spending and detailed visual analysis of infrastructure costs

Learn how these and other benefits of Ocean solve the top 6 challenges of optimizing Kubernetes infrastructure.



Table of contents

- 4** Challenge 1: Infrastructure management toil
 - 5** Challenge 2: Complexity & sprawl
 - 6** Challenge 3: Lack of transparency
 - 7** Challenge 4: Unpredictable costs & demand
 - 8** Challenge 5: Scalability
 - 9** Challenge 6: Balancing performance with efficiency
 - 10** Conclusion: A comprehensive Kubernetes optimization solution
- 

Challenge 1: Infrastructure management toil

Engineering teams are responsible for ensuring compute resources for their Kubernetes clusters are accurately and optimally provisioned. Unfortunately, day-to-day, practitioners often lack the time, skillset, and tooling needed to continuously adjust resources to best meet the ever-changing needs of applications. As a result, resources often go unoptimized, leading to an underutilization rate of 50% or higher for paid resources.

How Ocean eliminates this challenge

- Frees up more time for practitioners by automating otherwise time-intensive, complex tasks such as installing Kubernetes updates, rightsizing, and using spot instances
- Provides out-of-the-box automation templates for provisioning and scaling, ensuring continuous optimization across the board
- Eliminates manual work and enforces consistency via Infrastructure as Code (IaC) integrations.



Outcome: Infrastructure-related toil is drastically reduced, and practitioners can devote more time, energy, and mental capacity to high-value projects.



Ocean has helped customers improve IT staff efficiency by 41% and increase DevOps productivity by 30%.

Source: The Business Value of Spot by NetApp. IDC Research, 2022.

Challenge 2: Complexity & sprawl

The more dynamic and complex an infrastructure becomes, the more untenable it is for day-to-day practitioners to manually optimize resources for their Kubernetes containers. This is especially true for infrastructures that utilize multi-cloud environments.

How Ocean eliminates this challenge

- Uses virtual node groups (VNGs) to allow multiple infrastructure configurations on the same cluster, reducing the total number of node pools and clusters
- Automates bin packing, which reduces the number of nodes on each node group
- Anticipates and mitigates potential issues before they impact availability or performance



Outcome: Through heterogenous VNGs and automated bin packing, infrastructure sprawl is reduced to fewer node pools and clusters.



of respondents from Dimensional Research's survey still use manual methods to manage workload variation, and they cite the complexity of their cloud environments as their top challenge to achieving efficiency and automation.

Source: Optimizing in a Multi-Cloud World: Insights from Cloud Operations Professionals in 2024. Dimensional Research, 2024.

Challenge 3: Lack of cost transparency

A key priority for engineering teams is making sure mission-critical applications are always available and have the resources they need, regardless of cloud environment. Meanwhile, business teams need to understand and control the different IT cost centers associated with managing multi-cloud environments and using Kubernetes.

Both the IT and business teams must communicate to ensure mission-critical applications are running in the most cost-effective way possible. However, without unified insight into utilization rates and associated costs, stakeholders across departments lack the clarity to collaborate on spending, predict future expenditures, and make data-driven decisions that drive positive business outcomes.

How Ocean eliminates this challenge

- Provides granular visual cost analysis—including cost overviews, cost over time, and recommendations—so stakeholders can forecast cloud spend and identify waste
- Establishes accountability with real-time insight into costs by services, applications, users, and tasks
- Identifies potential cost savings across all major clouds



Outcome: Non-technical stakeholders have total visibility and a clear understanding of their Kubernetes infrastructure.



Challenge 4: Unpredictable costs & demand

Costs associated with cloud compute resources across multiple cloud vendors are hard to predict and impossible to forecast or budget. Moreover, the decentralized architecture of Kubernetes complicates usage tracking, leading to an even greater uncertainty in expenses.

Effective forecasting requires a solid understanding of the relationship between application needs, cluster performance, and resource allocation. Teams must consider factors like specific resource requirements for different applications, performance metrics of the cluster under various loads, and how resources are distributed across services. This presents a significant challenge for engineering teams that must balance performance with cost-efficiency. To manage unpredictability, teams often over-provision resources, inflating costs and limiting resource availability for other applications.

To overcome these challenges, teams must focus on implementing robust monitoring and analytics tools to track resource utilization effectively. Additionally, adopting agile methodologies and regularly reviewing and

adjusting capacity planning based on real-time data and user feedback can help teams adapt swiftly to changing requirements in dynamic Kubernetes environments.

How Ocean eliminates this challenge

- Combines granular analysis and IaC integrations for end-to-end optimization of a lean, scalable infrastructure
- Preemptively scales resources up or down to handle anticipated load changes before they occur
- Utilizes machine learning along with real-time and historical data to forecast future resource demands
- Supports labeling down to each node's individual capacities (e.g., CPU, memory), as well as network and storage cost analysis, enabling cost attributed to projects, teams, and business units for business-aligned forecasting



Outcome: Costs can be predicted and better managed via detailed insight into spending and optimization opportunities across every cloud environment.

Challenge 5: Scalability

An application's minute-to-minute resource demands can fluctuate wildly due to a whole host of factors, including user activity, backend processes, and even external events. Sudden spikes in usage can be hard to predict, leading developers with self-service resource allocation permissions to overprovision. Their goal is to keep applications available and responsive, but this can result in unintentional overspending on cloud resources, increasing operational costs that could be reduced with better resource management strategies. Balancing performance and cost efficiency is a key challenge in today's cloud-driven environment.

To tackle these challenges, teams should focus on implementing automated scaling mechanisms based on predefined metrics and thresholds to quickly adjust resource allocations and maintain optimal performance. Continuous monitoring, capacity planning, and performance tuning are crucial for anticipating demand spikes and refining resource provisioning strategies for better efficiency.

How Ocean eliminates this challenge

- Enables efficient resource allocation optimization with event-driven autoscaling
- Provides provisioning and autoscaling templates, enabling engineers to enforce scalable and cost-effective cloud usage
- Delivers application-aware, AI-driven rightsizing to adjusting resources to match the actual application needs, preventing both overprovisioning and underutilization



Outcome: Kubernetes workloads continue to run in any scenario without overprovisioning or overspending.



Challenge 6: Balancing performance with efficiency

Application performance and resource availability come at a price. To keep the cost of compute low for applications running in Kubernetes containers, Engineering teams must manually optimize resources, often across multiple clouds. However, for reasons already discussed, manually optimizing resources across multi-cloud environments is a time-intensive, energy-sapping endeavor.

Embracing a data-driven approach, leveraging monitoring tools to track key performance metrics, and iteratively refining resource allocation based on empirical data are crucial steps for harmonizing performance goals with operational efficiency in Kubernetes infrastructure.

How Ocean eliminates this challenge

- Delivers resource diversity mixing on-demand, commitments, and spot instances, as well as different machine sizes and architectures
- Provides comprehensive application-aware Kubernetes infrastructure management with AI/ML-driven autoscaling and automated rightsizing
- Enables the safe scaling of mission-critical Kubernetes workloads on discounted compute resources to reduce price
- Balances cost and availability through enhanced node autoscaling that accounts for application performance



Outcome: Cost, performance, and availability are balanced even in the most complex deployments across all major clouds.



Conclusion: A comprehensive Kubernetes optimization solution

Through AI/ML-driven optimization, Ocean ensures a continuously optimized Kubernetes infrastructure for cloud-native applications. Meanwhile, the entire Kubernetes infrastructure is automatically balanced for performance, availability, and cost across all major clouds. It removes manual optimization toil from IT staff and utilizes lightning-fast scaling to properly allocate resources without overprovisioning or risk of unplanned downtime.

Ocean in action

Experienced **74% savings** in compute costs



PORTER^o

Thanks to Spot by NetApp, cost optimization is now built into how we scale and manage our underlying infrastructure. Naturally, these cost savings are invested back in the business.”

– Jijo T. Joy, Senior DevOps Engineer at Porter

A serverless infrastructure engine for containers

Automate the optimization of your Kubernetes infrastructure across all major clouds—and dramatically lower your cloud costs while maintaining performance and availability:

[Request a demo and experience Ocean in action >>](#)

[Visit Spot Ocean to get started >>](#)