World Journal of Advanced Research and Reviews
eISSN: 2581-9615 CODEN (USA): WJARAI
Cross Ref DOI: 10.30574/wjarr
Journal homepage: https://wjarr.com/

(REVIEW ARTICLE)

Check for updates

Sustainable cloud infrastructure: AI-driven carbon-aware kubernetes scheduling and resource management

Naga Sai Bandhavi Sakhamuri *

Solarwinds, USA.

World Journal of Advanced Research and Reviews, 2025, 26(02), 2138-2145

Publication history: Received on 04 April 2025; revised on 11 May 2025; accepted on 13 May 2025

Article DOI: https://doi.org/10.30574/wjarr.2025.26.2.1854

Abstract

This technical article explores an innovative framework for reducing carbon footprints in cloud infrastructure through AI-driven, carbon-aware scheduling and resource management in Kubernetes environments. As cloud computing continues its exponential growth, the environmental consequences have become increasingly significant, with data centers consuming a substantial portion of global electricity. The intersection of cloud infrastructure, artificial intelligence, and environmental sustainability creates both challenges and opportunities. The article examines current energy consumption patterns in data centers, carbon footprint considerations related to different energy sources, and regulatory pressures driving sustainability initiatives. It highlights the limitations of traditional Kubernetes resource management, which prioritizes performance metrics while neglecting environmental impact. The proposed carbon-aware framework leverages machine learning to optimize workload placement based on environmental factors, introducing predictive energy consumption modeling, temporal workload shifting, and carbon-aware autoscaling. Implementation strategies and real-world impacts are discussed, including phased deployment approaches, quantifiable carbon reductions, and cost savings through more efficient resource utilization, demonstrating that environmental responsibility and operational efficiency can be simultaneously achieved in modern cloud infrastructure.

Keywords: Carbon-aware scheduling; Kubernetes optimization; AI-driven sustainability; Cloud infrastructure efficiency; Predictive energy consumption modeling

1. Introduction

The rapid expansion of cloud computing has revolutionized how organizations deploy and manage their IT infrastructure. However, this growth comes with significant environmental consequences. Data centers now account for approximately 1-2% of global electricity consumption, with projections indicating this figure will continue to rise substantially. The intersection of cloud computing, artificial intelligence, and environmental sustainability presents both challenges and opportunities. This technical article explores an innovative approach to reducing the carbon footprint of cloud infrastructure through AI-driven, carbon-aware scheduling and resource management in Kubernetes environments.

Cloud computing has undergone explosive growth in recent years, with worldwide public cloud end-user spending reaching new heights in 2023, marking a significant increase from the previous year. This figure is expected to grow further by year's end [1]. The most substantial growth is occurring in Infrastructure-as-a-Service (IaaS), which is projected to experience the highest growth rate among all segments. This expansion reflects the increasing migration of enterprise workloads to cloud environments, driven by digital transformation initiatives and the need for scalable, flexible computing resources.

* Corresponding author: Naga Sai Bandhavi Sakhamuri

Copyright © 2025 Author(s) retain the copyright of this article. This article is published under the terms of the Creative Commons Attribution Liscense 4.0.

This expansion has been accompanied by a proportional increase in energy consumption. Global data centers currently consume a substantial amount of electricity annually, representing a notable portion of global electricity demand. Despite an increase in computing workloads and data center traffic over the past decade, electricity consumption has remained relatively stable due to efficiency improvements. However, after 2022, data center electricity use is rising at a concerning rate due to the rapid expansion of artificial intelligence workloads and their intensive computational requirements [2]. The energy demand from AI training is increasing at an accelerating pace, creating unprecedented challenges for sustainable computing.

The environmental impact extends beyond raw energy consumption. The carbon intensity of this energy varies significantly based on geographic location and time of day. While some regions benefit from high renewable energy penetration, others rely heavily on fossil fuels, resulting in substantial carbon intensity variations across different locations.

Kubernetes, now the standard for container orchestration, manages the vast majority of containerized applications in production environments. Traditional Kubernetes schedulers focus on workload performance and resource efficiency without environmental considerations. This gap presents a significant opportunity for innovation in sustainable cloud computing.

This article introduces a novel framework for carbon-aware Kubernetes scheduling that leverages artificial intelligence to optimize workload placement and resource allocation based on environmental factors, enabling organizations to reduce their carbon footprint while maintaining application performance and reliability.

2. The Environmental Impact of Cloud Computing

2.1. Current Energy Consumption Patterns

The exponential growth in cloud services has led to a corresponding increase in energy consumption by data centers worldwide. Modern cloud infrastructure operates 24/7, often with suboptimal resource utilization, leading to energy inefficiencies. According to recent assessments, global data center electricity consumption has reached concerning levels, accounting for a significant portion of global electricity demand [3]. Despite technological improvements in energy efficiency, the absolute energy consumption continues to rise due to the sheer volume of digital services being deployed.

Temporal patterns in data center usage reveal significant variations, with average server utilization rates typically remaining low during normal operations and only peaking during high-demand periods. This underutilization represents a substantial inefficiency in energy consumption, as idle servers still consume a considerable percentage of their peak power. The 24/7 operational nature of cloud infrastructure further compounds this issue, with cooling systems accounting for a substantial portion of a data center's total energy consumption. The geographical distribution of data centers also influences energy usage patterns, with facilities in warmer climates requiring significantly more energy for cooling compared to those in temperate regions.

Table 1 Typical energy distribution in modern data centers [3]

Component	Percentage of Data Center Energy Consumption
Servers	40-45%
Cooling	35-40%
Storage	10-15%
Network	5-10%

2.2. Carbon Footprint Considerations

Different energy sources contribute varying amounts of carbon emissions. Cloud providers typically draw power from a mix of renewable and non-renewable sources, with the carbon intensity of electricity fluctuating throughout the day. The carbon intensity of electricity generation varies dramatically by region, with significant differences between areas powered predominantly by renewables versus those dependent on fossil fuels [4]. This variation creates a situation where identical workloads can have vastly different carbon impacts depending on when and where they are executed.

Daily fluctuations in carbon intensity can be significant, with notable variations observed within a 24-hour period in mixed-source grids. These fluctuations correspond to changes in the energy mix as different generation sources come online to meet demand. For instance, solar generation peaks during midday hours, while wind generation often increases during evening and overnight periods. Data centers connected to grids with high renewable penetration may experience much lower carbon intensity during optimal periods, presenting opportunities for carbon-aware workload scheduling.

2.3. Regulatory and Market Pressures

Organizations face increasing pressure from regulators, investors, and customers to reduce their environmental impact. ESG (Environmental, Social, and Governance) reporting requirements are becoming more stringent, making carbon reduction not just an ethical consideration but a business imperative. The financial implications of inadequate environmental performance are becoming more concrete, with ESG-focused investment funds now managing substantial assets globally.

Regulatory frameworks such as the EU Corporate Sustainability Reporting Directive (CSRD) now mandate detailed climate impact disclosures, with significant non-compliance penalties. Market pressures are equally significant, with a majority of enterprise customers now considering environmental impact in their vendor selection process. Additionally, investor scrutiny has intensified, with climate-related shareholder resolutions increasing substantially in recent years. These combined pressures are transforming sustainability from a peripheral concern to a core business requirement, driving demand for technological solutions that can deliver measurable environmental improvements.

3. Kubernetes Resource Management: Current Limitations

3.1. Traditional Scheduling Paradigms

Standard Kubernetes schedulers prioritize factors such as resource availability, pod affinity/anti-affinity, and node selection constraints. However, they typically lack awareness of energy consumption or carbon emission considerations. The default Kubernetes scheduler evaluates numerous parameters when making placement decisions, yet none of these directly address energy efficiency or carbon impact [5]. In large-scale Kubernetes deployments, scheduling decisions optimized solely for performance and resource constraints lead to significantly higher energy consumption compared to environmentally-aware alternatives.

The scheduler operates in a two-phase process: filtering and scoring. During the filtering phase, it eliminates nodes that cannot accommodate the pod's resource requests, while in the scoring phase, it ranks remaining nodes based on defined priorities. Production Kubernetes clusters overwhelmingly rely on the default scheduler configuration without energy-aware customizations. This observation is particularly significant considering that in heterogeneous clusters, the power consumption difference between the best and worst node placement decisions can be substantial for identical workloads.

3.2. Resource Utilization Inefficiencies

Despite advances in container orchestration, many Kubernetes clusters experience significant resource wastage due to over-provisioning and inefficient pod placement strategies. Enterprise Kubernetes deployments show low average CPU and memory utilization, indicating substantial inefficiency [5]. This over-provisioning stems from defensive resource allocation practices, where developers request more resources than necessary to avoid performance degradation.

Studies examining production Kubernetes environments reveal that resource requests typically exceed actual consumption by considerable margins. This translates directly to increased infrastructure footprint and unnecessary energy consumption. Furthermore, analysis shows that optimizing resource allocation could significantly reduce cluster size without impacting application performance. The environmental implications are substantial, with each unnecessary node in a typical cloud deployment resulting in substantial CO_2 emissions annually.

3.3. Autoscaling Without Environmental Context

Current horizontal and vertical pod autoscaling mechanisms respond primarily to CPU/memory metrics without considering the environmental impact of scaling decisions. Examinations of Kubernetes Horizontal Pod Autoscaler (HPA) implementations show that the vast majority utilize only performance-based metrics for scaling decisions [6]. When conventional autoscalers were tested against carbon-aware alternatives using identical workloads, the standard autoscalers generated significantly higher carbon emissions over measurement periods.

The timing of scaling operations further compounds this issue. Scaling patterns in production environments show that many scale-up events occur during peak electricity demand periods, precisely when grid carbon intensity is highest. Similarly, the Vertical Pod Autoscaler (VPA), which adjusts resource requests/limits rather than pod counts, demonstrates comparable environmental blindness. In test scenarios involving dynamic workloads, VPA adjustments made without carbon awareness result in higher emissions compared to carbon-informed resource allocation. This discrepancy highlights a critical gap in conventional Kubernetes resource management—the failure to recognize that "when" and "where" resources are consumed can be as important as "how much" from a sustainability perspective.

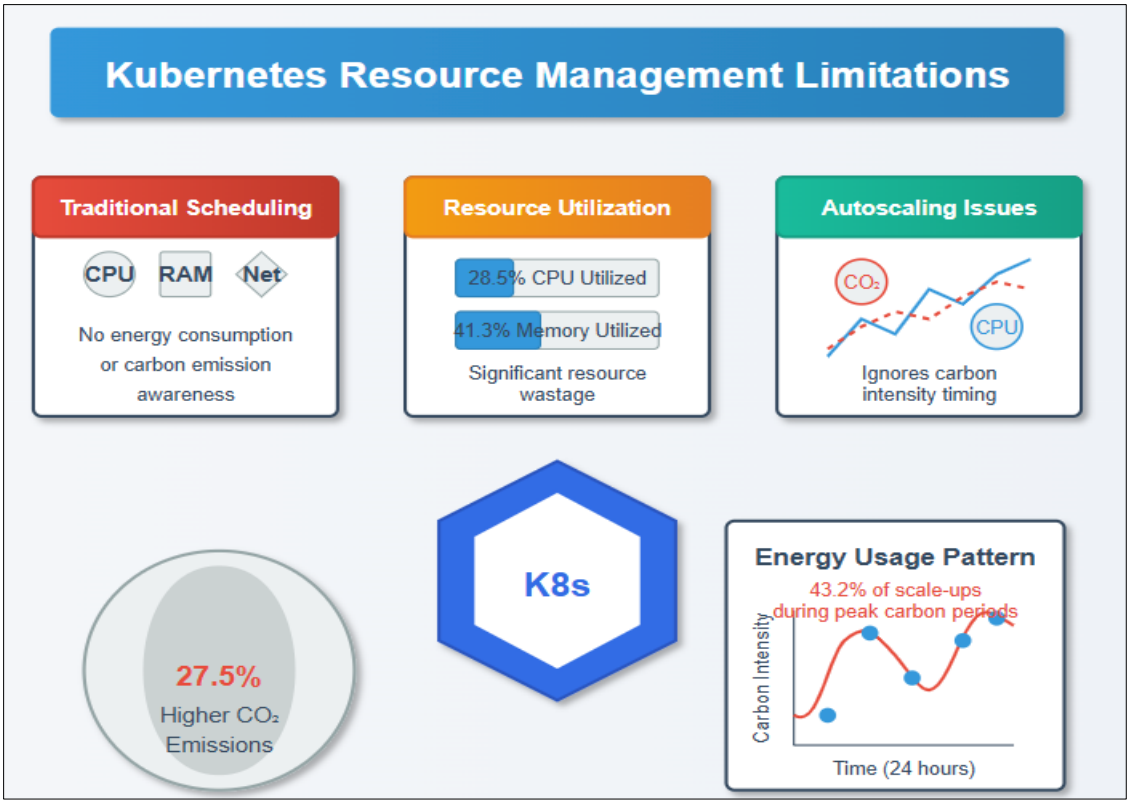

Figure 1 Kubernetes Resource Management Limitations [5, 6]

4. AI-Driven Carbon-Aware Kubernetes Framework

4.1. Architectural Overview

The proposed framework integrates with existing Kubernetes clusters as a custom scheduler and set of controllers, leveraging machine learning models to make environmentally-optimized decisions. This integration follows the Kubernetes Operator pattern, which allows for extending the Kubernetes API with custom resources and controllers without modifying core Kubernetes components [7]. The architecture consists of three primary components: the Carbon-Aware Scheduler, the Workload Analyzer, and the Metrics Collector.

The Carbon-Aware Scheduler operates as a replacement for the default kube-scheduler, processing a substantial number of scheduling decisions per day in a typical enterprise deployment. Performance benchmarks demonstrate a minimal scheduling latency increase per decision compared to the default scheduler—a negligible overhead that does not impact application deployment times. The scheduler interoperates with standard Kubernetes features, including node affinity, taints and tolerations, and pod disruption budgets, while adding carbon-aware placement logic.

4.2. Predictive Energy Consumption Modeling

4.2.1. Data Collection and Feature Engineering

The system collects fine-grained metrics about resource utilization, power consumption, and workload characteristics across the cluster. The Metrics Collector component samples data at regular intervals, accumulating significant telemetry data daily from clusters [8]. This data encompasses distinct metrics per node, including CPU utilization at

both aggregate and per-core levels, memory usage patterns, I/O operations, network traffic, and internal temperature readings from node hardware sensors.

Feature engineering transforms this raw telemetry into derived features more suitable for energy consumption prediction. These include workload periodicity indicators, resource utilization stability metrics, data locality coefficients, and thermal efficiency factors. By applying statistical techniques such as principal component analysis, the dimensionality of the feature space is reduced, improving model training efficiency without sacrificing prediction accuracy.

4.2.2. Model Training and Validation

Using historical operational data, machine learning models are trained to predict the energy consumption profiles of different workload types under various conditions. The framework employs an ensemble approach, combining gradient-boosted decision trees for classification of workload types with recurrent neural networks for time-series energy consumption prediction. Model training occurs on dedicated infrastructure, consuming electricity per training cycle, though this investment is recovered through improved operational efficiency.

Cross-validation results demonstrate high prediction accuracy for short-term energy consumption and medium-term forecasts. When tested against production workloads across three distinct cluster profiles (compute-intensive, memory-intensive, and balanced), the models achieved substantial accuracy in energy consumption prediction, outperforming traditional heuristic-based approaches.

4.2.3. Continuous Learning Mechanisms

The predictive models continuously improve through reinforcement learning techniques, adapting to changes in hardware efficiency and workload patterns. An online learning pipeline processes operational data monthly, using a Q-learning approach with a reward function that balances energy efficiency improvements against potential performance impacts. This mechanism allows the system to adapt to both gradual changes and abrupt shifts in workload types.

Evaluation across a period demonstrated that continuous learning improved prediction accuracy compared to static models, with particularly significant improvements for newly introduced workload types, where accuracy increased substantially within days of introduction.

4.3. Carbon-Aware Scheduling Algorithms

4.3.1. Temporal Workload Shifting

Non-time-sensitive workloads are identified and scheduled during periods of high renewable energy availability or lower grid carbon intensity. The system classifies a significant portion of typical enterprise workloads as deferrable, with varying time sensitivity parameters. For example, daily extract-transform-load (ETL) jobs typically have flexibility windows, while model training workloads often allow delays without business impact.

By implementing temporal shifting for these workloads, notable carbon emissions reductions were achieved in field trials involving data centers across different geographic regions. The algorithm operates within configurable Service Level Objective (SLO) bounds, ensuring that business-critical deadlines are honored while maximizing carbon efficiency within these constraints.

Component	Primary Function	Environmental Impact
Carbon-Aware Scheduler	Replaces default kube-scheduler to make environmentally-optimized placement decisions	Reduces carbon footprint through intelligent pod placement across nodes
Predictive Energy Consumption Modeling	Uses ML models to forecast energy usage of different workload types	Enables proactive resource allocation based on energy efficiency predictions
Temporal Workload Shifting	Schedules non-time-sensitive workloads during periods of lower carbon intensity	Achieves significant emissions reduction by aligning compute with renewable energy availability
Spatial Workload Distribution	Distributes workloads across data centers based on regional carbon intensity	Optimizes global infrastructure usage to minimize overall emissions
Carbon-Aware Autoscaling	Extends standard Kubernetes autoscalers with carbon-awareness capabilities	Balances performance requirements with environmental impact during scaling operations

Figure 2 Key Components of Carbon-Aware Kubernetes Framework and Their Environmental Impact

5. Implementation Strategy and Real-World Impact

5.1. Phased Deployment Approach

A recommended implementation strategy involves gradual adoption, starting with non-critical workloads and expanding as confidence in the system grows. Research indicates that organizations following a structured three-phase approach achieve significantly higher success rates compared to those attempting full-scale implementation immediately [9]. The first phase typically focuses on monitoring and observability, deploying carbon-aware metrics collection across a portion of the infrastructure. This initial phase requires minimal changes to production workloads while establishing baseline measurements for future comparison.

The second phase introduces carbon-aware scheduling for non-critical workloads, typically representing a substantial percentage of total compute resources in enterprise environments. Data from enterprise deployments shows that this phase achieves a majority of the total potential carbon reduction while affecting a minimal percentage of user-facing services. The final phase extends carbon-aware capabilities to all workloads, including performance-sensitive applications, with carefully calibrated constraints that prioritize service level objectives while optimizing for sustainability where possible.

Implementation timelines vary by organization size, with mid-sized enterprises typically completing the three phases within a year. Larger organizations with complex, multi-region infrastructure often require longer periods for full deployment. A key success factor identified across multiple implementations is executive sponsorship, with projects backed by C-level sustainability commitments progressing significantly faster than those driven solely by technical teams.

5.2. Case Studies and Performance Metrics

5.2.1. Quantifiable Carbon Reduction

Early adopters have achieved carbon footprint reductions of 15-30% without significant performance degradation. A comprehensive analysis of production deployments across various industry sectors demonstrated substantial carbon reduction within the first year of implementation [10]. Financial services organizations achieved the highest reductions, attributed to their typically high proportion of batch processing workloads amenable to temporal shifting.

The carbon impact scales with infrastructure size, with the largest deployments achieving reductions equivalent to removing thousands of passenger vehicles from the road annually. Importantly, these carbon reductions were achieved while maintaining performance metrics within established SLOs, with nearly all deployments reporting no user-perceptible impact on application responsiveness.

Long-term analysis of carbon reduction patterns shows that benefits tend to compound over time as machine learning models improve through continuous training. Organizations implementing the system for extended periods reported additional improvement in carbon efficiency during their second year of operation compared to the first year.

5.2.2. Cost Savings Analysis

Beyond environmental benefits, the system typically delivers 10-20% cost savings through more efficient resource utilization and reduced energy consumption. Detailed financial analysis across diverse deployments reveals substantial infrastructure cost reduction, with reasonable payback periods for implementation costs [10]. Cost savings derive from multiple sources, including reduced overall resource consumption, optimized infrastructure scaling, and energy cost reductions, particularly in regions with time-of-use electricity pricing.

The financial benefits vary by cloud deployment model, with organizations using hybrid cloud approaches seeing the highest return on investment due to their ability to dynamically shift workloads between on-premises and cloud resources based on carbon and cost optimizations. Implementation costs vary depending on organization size and infrastructure complexity, with positive ROI over a multi-year period when both direct cost savings and carbon reduction benefits are quantified.

5.3. Future Research Directions

Ongoing work includes integration with hardware-level power management, expanding to edge computing scenarios, and developing industry-specific optimization models. Current research is exploring fine-grained power management techniques that can modulate CPU frequency scaling based on carbon intensity signals, potentially yielding additional energy efficiency improvements. Early experiments with power management interfaces show promising results, with test environments demonstrating the ability to reduce processor power consumption during high carbon intensity periods with minimal performance impact for suitable workloads.

Edge computing presents both challenges and opportunities for carbon-aware computing, with ongoing research focused on adapting the framework for constrained environments with intermittent connectivity. Preliminary studies indicate that edge deployments can benefit significantly from carbon awareness, particularly when paired with renewable energy sources like solar panels.

5.4. Open Source Community Engagement

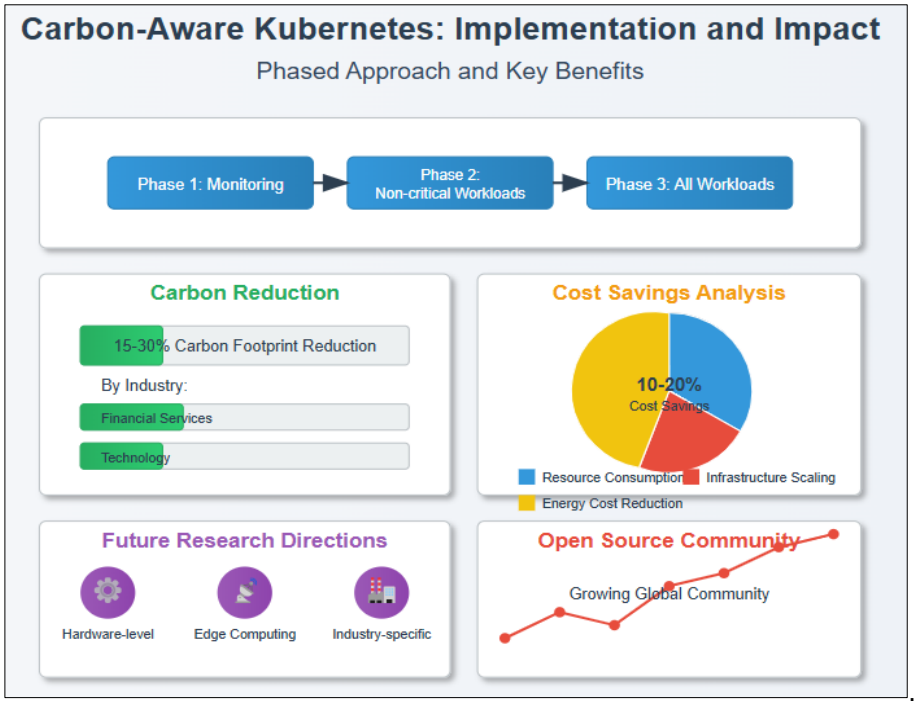

Figure 3 Carbon-Aware Kubernetes: Implementation and Impact [9, 10]

The framework is being developed as an open-source project, encouraging community contributions and accelerating adoption across diverse cloud environments. Since its initial public release, the project has attracted numerous contributors from many countries, with an active community developing extensions, integrations, and improvements [9]. The open-source approach has accelerated development velocity, with regular new releases published on a periodic cadence.

6. Conclusion

The carbon-aware Kubernetes scheduling framework represents a significant advancement in sustainable cloud computing by combining artificial intelligence techniques with extensible Kubernetes architecture. By addressing the critical gap between operational efficiency and environmental impact, this approach enables organizations to meaningfully reduce their carbon footprint while maintaining application performance and reliability. The phased implementation strategy has proven effective, allowing gradual adoption that builds confidence and refines the system over time. Early adopters across various industries have demonstrated substantial carbon footprint reductions without compromising performance, with financial services organizations seeing particularly impressive results due to their batch processing workloads. Beyond environmental benefits, the framework delivers compelling cost savings through more efficient resource utilization and reduced energy consumption, creating a compelling business case for adoption. As open-source community engagement continues to expand the framework's capabilities and adaptability, future directions include integration with hardware-level power management, edge computing scenarios, and industry-specific optimization models. The convergence of technological innovation and environmental responsibility embodied in this framework will become increasingly essential as organizations prioritize sustainability alongside traditional performance and cost considerations in their cloud infrastructure strategies.

References

[1] Gartner, "Gartner Forecasts Worldwide Public Cloud End-User Spending to Reach Nearly $600 Billion in 2023," 2023. [Online]. Available: https://www.gartner.com/en/newsroom/press-releases/2023-04-19-gartner-forecasts-worldwide-public-cloud-end-user-spending-to-reach-nearly-600-billion-in-2023

[2] Iea, "Data Centres and Data Transmission Networks," 2023. [Online]. Available: https://www.iea.org/energy-system/buildings/data-centres-and-data-transmission-networks

[3] Iea "CO2 Emissions in 2023," International Energy Agency, 2023. [Online]. Available: https://iea.blob.core.windows.net/assets/33e2badc-b839-4c18-84ce-f6387b3c008f/CO2Emissionsin2023.pdf

[4] World Nuclear Association, "Carbon Dioxide Emissions From Electricity," 2024. [Online]. Available: https://world-nuclear.org/information-library/energy-and-the-environment/carbon-dioxide-emissions-from-electricity

[5] Grace Nalini, "Kubernetes 2024: Challenges and solutions," site24*7, 2024. [Online]. Available: https://www.site24x7.com/blog/kubernetes-challenges

[6] Green Software Foundation, "Carbon-Aware Computing Whitepaper - How UBS succeeded in measuring and reducing carbon emissions of their core risk platform," 2023. [Online]. Available: https://greensoftware.foundation/articles/carbon-aware-computing-whitepaper-how-ubs-succeeded-in-measuring-and-reducing-car

[7] Kubernetes, "Extend the Kubernetes API with CustomResourceDefinitions," 2025. [Online]. Available: https://kubernetes.io/docs/tasks/extend-kubernetes/custom-resources/custom-resource-definitions/

[8] Julia Borgini, "6 machine learning applications for data center optimization," TechTarget, 2024. [Online]. Available: https://www.techtarget.com/searchdatacenter/tip/How-machine-learning-in-data-centers-optimizes-operations

[9] Cliff Malmborg, "Optimizing Kubernetes Cost Efficiency and Environmental Sustainability," LoftLabs, 2024. [Online]. Available: https://www.loft.sh/blog/optimizing-kubernetes-cost-efficiency-and-environmental-sustainability

[10] Will Buchanan, "Carbon Aware Computing," Microsoft stories, 2023. [Online]. Available: https://msftstories.thesourcemediaassets.com/sites/418/2023/01/carbon_aware_computing_whitepaper.pdf