

A Comprehensive Survey of Micro Datacenter: Current Technologies and Future Possibilities

Jinyang GUO, Mingxuan ZHANG, Yunwei LI, Chao LI(✉), Minyi GUO

School of Computer Science, Shanghai Jiao Tong University, Shanghai 200240, China

© Higher Education Press 2025

Abstract

Micro datacenters (μ DCs) are emerging miniaturized datacenters poised to revolutionize distributed computing in edge environments by seamlessly integrating compact, modular architectures with localized computing, storage, and networking capabilities. This survey provides a complete and up-to-date review of μ DC. We introduce a comprehensive three-layer analytical framework essential for understanding and evaluating μ DC technologies: 1) The infrastructure layer empowers μ DCs to adapt physical resources, including compute units; 2) The platform layer prioritizes middleware-driven resource orchestration, driving improvements in computational efficiency and energy optimization; 3) At the application layer, μ DCs are committed to delivering performance optimizations that meet demanding requirements for latency, reliability, and energy efficiency. This work synthesizes leading-edge implementations and technology trends, clearly outlining critical trade-offs, identifying significant challenges, and defining the research frontiers vital for advancing

μ DC design and deployment across diverse industrial and extreme environments.

Keywords Micro Datacenter, Distributed Collaboration, Edge Computing

1 Introduction

The rapid proliferation of edge computing paradigms and growing demands for localized computation have driven significant shifts in datacenter infrastructure. As traditional centralized datacenters struggle with limitations in latency, bandwidth, and situational adaptability, the concept of μ DCs has emerged as a powerful alternative to meet the challenges presented by edge-intensive applications. μ DCs are decentralized computing nodes designed to offer localized compute, storage, and network capabilities near data sources and users. Their compact, modular, and self-contained nature makes them ideally

Received month dd, yyyy; accepted month dd, yyyy

E-mail: lichao@cs.sjtu.edu.cn

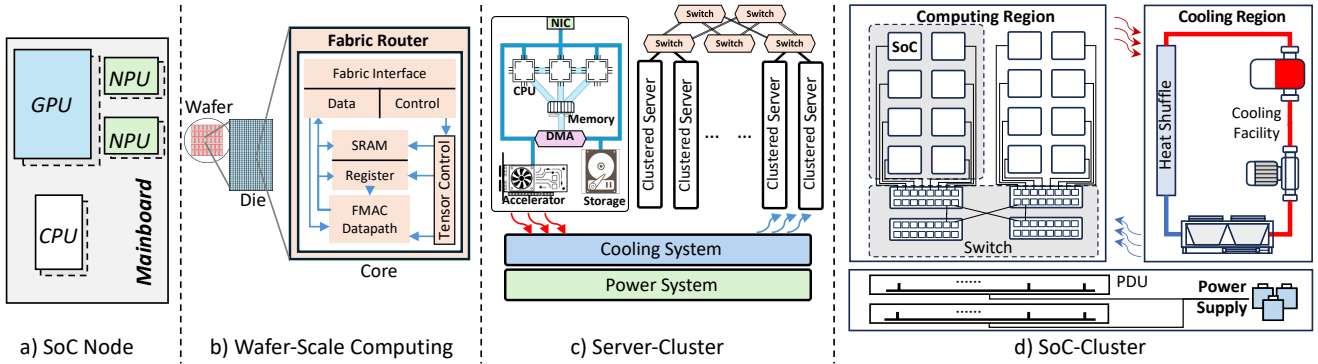


Fig. 1 An overview of promising μ DC architectures

suitable for a wide range of scenarios, from autonomous vehicular systems [1–7] and industrial IoT [8–10] to underwater infrastructures [11–15] and space missions [16–22].

Despite the growing relevance and deployment of μ DCs, the field remains in a state of evolution, characterized by considerable heterogeneity in infrastructure configurations, operational mechanisms, and application objectives. Consequently, understanding the full potential and limitations of μ DCs requires a structured and comprehensive analysis across multiple layers. To systematically address this complexity, this survey investigates μ DCs through three critical perspectives: Infrastructure, Platform, and Application, each addressing a fundamental overarching question central to the advancement of μ DC technologies.

From an infrastructure perspective, we explore how the physical infrastructure designs of μ DCs respond to diverse operational constraints posed by various environments. As shown in Figure 1, several promising architectures exist that involve examining compute and storage architectures, network connectivity models, energy and power solutions, and cooling technologies tailored to address specific contextual challenges, such as limited physical space, environmental extremes, and resource constraints. By analyzing real-world case

studies, the infrastructure section highlights critical trade-offs and innovative strategies employed to ensure reliability, efficiency, and adaptability across varied deployments.

From the Platform perspective, the survey examines how μ DC operational platforms effectively orchestrate hardware and software resources to strike a balance between performance, energy efficiency, and resilience. Here, we delve into the middleware and operational mechanisms enabling μ DCs to dynamically manage resource allocation, adapt to fluctuating workloads, and sustain robust operations under challenging conditions. Detailed examinations of system types, configuration paradigms, design overhead considerations, and resource management strategies provide insights into the design patterns that govern the complexity and scalability inherent in μ DC operations.

At the Application level, the focus shifts to understanding how μ DCs optimize their capabilities to satisfy diverse, context-specific demands at the edge. This involves analyzing how μ DC architectures are adapted to distinct application scenarios, including latency-critical tasks such as autonomous driving and robotics, reliability-critical operations in subsea and space deployments, and energy-constrained scenarios in remote microgrid contexts. Through examining μ DC performance in

balancing metrics such as latency, throughput, accuracy, and energy efficiency, the application-level analysis demonstrates the effectiveness of μ DCs in optimizing computational workloads across various spatial, temporal, and operational scopes.

This survey contributes to the existing body of knowledge in several ways. First, it provides a structured and comprehensive classification of μ DC technologies based on real-world deployments, systematically distinguishing among key infrastructural configurations, operational strategies, and application-driven optimizations. Second, by leveraging an integrated analytical framework organized around infrastructure, platform, and application layers, this work offers actionable insights into the complex interdependencies and trade-offs inherent in μ DC design and operation. Lastly, the identification and synthesis of current technological trends, along with clear discussions on remaining challenges and potential future directions, provide valuable guidance for both researchers and practitioners seeking to advance μ DC capabilities.

The remainder of this paper is organized as follows: Section 2 provides an overview of μ DC concepts, typologies, and key features. Sections 3, 4, and 5 provide detailed analyses of the infrastructure, platform, and application layers, respectively, accompanied by in-depth case studies and comparative discussions. Section 6 discusses the main opportunities and challenges. Finally, Section 7 concludes by summarizing key findings and outlining future research directions.

2 Micro datacenter: An Overview

Perspectives aside, it is logical for the community to maintain diverse definitions and viewpoints on micro datacenters while the field continues to evolve and technical maturity is still forthcoming.

In this formative stage, all contributions carry reasonable value and merit systematic discussion. To thoroughly understand the roles and capabilities of μ DCs, it is essential to examine them through a structured lens that incorporates their underlying infrastructure, operational mechanisms, and application scenarios. As shown in Table 1, this section first categorizes representative μ DC deployments and then systematically contrasts them across three fundamental layers, including infrastructure, platform, and application.

2.1 Definition on μ DC

μ DCs represent a decentralized computing paradigm that complements cloud and centralized datacenter architectures by delivering localized compute, storage, and networking services in close physical proximity to end users or data-generating sources. Unlike hyperscale or regional facilities, μ DCs [23, 24] are typically compact, modular, and often purpose-built for autonomous or semi-autonomous operations in constrained or dynamic environments. The need for low-latency processing, edge intelligence, and situational autonomy in bandwidth-limited or mission-critical scenarios drives this architectural decentralization.

μ DCs are specifically designed to support distributed edge computing by processing data locally. This approach reduces latency, decreases network congestion, and improves responsiveness. Architecturally, μ DCs combine essential IT resources, including compute units and storage, with vital subsystems for power, cooling, and environmental protection. These systems are often customized for use in challenging environments, such as vehicles, offshore sites, satellites, or remote microgrids.

μ DCs are distinguished from generic edge computing infrastructure by four defining characteris-

Table 1 The overview of μ DC

<div>Features</div> <div>Typologies</div>	Infrastructure						Platform					Application			
	Modular rack design?	Abundant compute/storage?	Robust network capacity?	Shared resource pool?	Sufficient power supply?	Effective cooling system?	Resource mgmt./virtualization?	Autonomous/ self- mgmt.?	Collaborative integration?	Mobility/Portability?	Support remote control?	Latency-sensitive apps?	Best-effort apps?	Support mix-load apps?	Customizable service?
<i>Vehicular μDC</i>	○	◐	○	◐	◐	◐	●	●	◐	●	●	●	●	●	◐
<i>IIoT μDC</i>	●	◐	●	●	●	●	●	●	●	◐	●	●	◐	●	●
<i>MEC Units</i>	●	●	●	●	●	●	●	●	●	◐	●	●	●	◐	◐
<i>Microgrid μDC</i>	◐	◐	◐	◐	◐	●	●	●	●	○	●	◐	◐	◐	●
<i>Underwater μDC</i>	●	●	●	●	●	●	●	●	●	○	●	○	●	○	●
<i>Space μDC</i>	○	◐	◐	◐	○	○	●	●	◐	●	◐	○	●	○	●

● -Yes ○ -No ◐ -Possible

tics. First, their deployability is enabled by pre-integrated, containerized architectures that facilitate rapid installation in resource-limited or mobile scenarios [24]. Second, operational autonomy is achieved through embedded self-diagnostic capabilities, which support automated fault recovery, intelligent energy management, and remote orchestration—capabilities essential for unmanned operations [25]. Third, environmental resilience is paramount due to atypical deployment environments, including underwater installations like Microsoft Natick [13], spaceborne platforms such as HPE’s Spaceborne Computer-2 [26] and Space Microdatacenter [16], and vehicular integrations like NVIDIA’s automotive edge systems [1, 27]. These scenarios demand robust tolerance to thermal fluctuations, mechanical shocks, and unstable power conditions. Finally, μ DCs feature architectural scalability, operating both as standalone micro-nodes and as federated clusters spanning ge-

ographic regions, from nano-clusters for remote IoT sensing [28] to large-scale deployments within urban smart grid infrastructure.

Unlike centralized cloud datacenters, which are optimized for global resource pooling, workload consolidation, and economies of scale, μ DCs prioritize spatial locality, tight vertical integration, and operational autonomy. Their edge-centric placement makes them well-suited to latency-critical workloads such as edge-native AI inference, industrial control, emergency response, and intermittently connected analytics. Although related to edge, fog, and cloudlet paradigms, μ DCs differ in their concrete embodiment and system focus: fog denotes architectural layering and workload placement across the cloud–edge continuum, and cloudlets emphasize localized compute for mobile users. By contrast, μ DCs are physically bounded, infrastructure-defined nodes that deliver services autonomously or semi-autonomously under non-

Table 2 The abbreviation used in this survey

Abbr.	Full Term
μ DC	Micro Data Center
MEC	Multi-access Edge Computing
IIoT	Industrial Internet of Things
SoC	System on Chip
VM	Virtual Machine
STSU	Single Task, Single User
STMU	Single Task, Multiple Users
MTSU	Multiple Tasks, Single User
MTMU	Multiple Tasks, Multiple Users
PDU	Power Distribution Unit
NIC	Network Interface Card
FPGA	Field Programmable Gate Array
NPU	Neural Processing Unit
GPU	Graphics Processing Unit
CPU	Central Processing Unit
DMA	Direct Memory Access
DNN	Deep Neural Network
BE	Best-Effort
LC	Latency-Critical
RISC	Reduced Instruction Set Computer
UPS	Uninterruptible Power Supply
AR/VR	Augmented Reality / Virtual Reality
QoS	Quality of Service
LPWAN	Low-Power Wide-Area Network

ideal field conditions.

In this survey, we adopt a broad yet principled scope: an μ DC is defined as any logically autonomous, physically bounded, and infrastructure-constrained data processing system deployed in spatial separation from core cloud facilities. It must provide computing, storage, and network capabilities with a degree of self-containment, situational adaptability, and operational granularity appropriate for edge-intensive workloads. To facilitate comprehension, we have compiled all abbreviations used in this work in the Table 2.

2.2 μ DC Typologies

The wide range of micro datacenter deployments highlights the diverse requirements of edge computing across various sectors, including industry, transportation, urban infrastructure, and remote environments. To systematically characterize this diversity, we categorize μ DCs into six primary

types based on the deployment. These typologies span environments from highly mobile and energy-constrained sites to harsh underwater and orbital scenarios, each presenting unique technical challenges and innovation opportunities. This framework facilitates a comprehensive and comparative understanding of μ DC technologies, informing future research directions.

Vehicular μ DC [1–7]: Deployed within mobile platforms such as autonomous vehicles, public transit systems, and unmanned aerial vehicles, Vehicular μ DCs are optimized for high mobility and real-time perception-compute-actuation loops. These systems are engineered to deliver ultra-low latency inference, multi-modal sensor fusion, and autonomous decision-making under stringent power and thermal constraints.

IIoT μ DCs [8–10]: Engineered for deployment within factories, refineries, or other mission-critical industrial environments, IIoT μ DCs emphasize real-time control, edge analytics, and high availability. They typically feature ruggedized enclosures, redundant networking, and integration with industrial control systems.

MEC Units [29–35]: Co-located with cellular base stations or metro network edges, MEC Units serve as low-latency offloading nodes for telecom networks, enabling AR/VR, video streaming, and mobile AI workloads.

Microgrid μ DCs [36–40]: These μ DCs function in energy-constrained or off-grid settings and integrate with localized renewable or hybrid power systems, including solar, batteries, and diesel. These μ DCs offer enhanced energy autonomy, scalable storage, and efficient thermal solutions.

Underwater μ DCs [11–15]: Installed on the waterfloor, these μ DCs address physical security, thermal management, and latency requirements for

maritime or coastal data workloads. These μ DCs leverage passive liquid cooling, sealed containerization, and remote-controlled diagnostics.

Space μ DCs [16–22]. Designed for operation in orbital or extraterrestrial environments, Space μ DCs must withstand radiation, vacuum, extreme temperature variation, and limited communication windows. These systems are deeply autonomous, fault-tolerant, and energy-optimized.

2.3 Key Features

The inherent operational complexity and contextual heterogeneity of μ DCs demand a systematic framework for analyzing their architectural paradigms, operational behaviors, and application-specific requirements. This survey proposes a tripartite analytical model that decomposes μ DC ecosystems into three interdependent strata, including underlying infrastructure, operating platform, and application scenarios. This stratification enables comprehensive system characterization, facilitates insights into cross-layer optimization, and supports generalized comparisons across diverse μ DC implementations.

Infrastructure. As detailed in Section 3, this foundational stratum governs the physical and resource constraints defining μ DC capabilities. Its composition spans four critical domains: computational units paired with storage architectures that collectively dictate workload compatibility and processing reliability; network infrastructure differentiating between wired and wireless connectivity paradigms with distinct bandwidth-latency profiles that determine edge device interoperability; energy provisioning systems balancing grid dependence, renewable integration, and backup solutions to ensure operational continuity in resource-constrained environments; and thermal manage-

ment architectures employing air-based, liquid-cooled, or spatial heat dissipation strategies that intrinsically couple with hardware form factors and energy budgets. These elemental components establish the physical boundaries for scalability, deployment flexibility, and operational sustainability.

Platform. Section 4 examines this middleware stratum, which orchestrates the utilization of hardware resources through software-defined control mechanisms. Three pivotal operational dimensions emerge: virtualization architectures differentiate between bare-metal, VM-based, and containerized implementations, with corresponding performance-security tradeoffs; configuration methodologies employ static, dynamic, or hybrid control paradigms to adapt to fluctuating workload demands and environmental variations. This platform layer functions as the intelligent intermediary, translating physical infrastructure capabilities into dynamically allocatable service resources.

Application. The operational validity of μ DCs, culminating in their capacity to execute latency-sensitive, autonomous workloads under stringent performance constraints, as analyzed in Section 5. Application profiles manifest through four discriminative characteristics: service criticality spectrum ranging from best-effort to mission-critical task execution; optimization vectors prioritizing throughput maximization, energy conservation, or inference accuracy enhancement; spatiotemporal constraints dictating geographic distribution patterns and real-time response requirements; multi-scale optimization techniques spanning from data-layer model compression to system-wide orchestration, each introducing distinct operational tradeoffs between computational efficiency and result fidelity. These application-driven parameters ultimately determine resource allocation strategies while expos-

ing inherent limitations in synchronization overhead and accuracy degradation.

We begin with typology-specific practical priorities to ground the discussion before abstracting metrics into a general framework. Performance metric priorities vary across micro datacenter typologies. Vehicular μ DCs emphasize ultra-low tail latency, seamless mobility continuity, and deterministic isolation for mixed-criticality workloads, while also prioritizing energy efficiency in accelerator-rich configurations [3,5,6]. IIoT μ DCs prioritize very high availability, tightly bound jitter for control stability, conservative thermal safety margins, and predictable energy profiles over sustained operation [10]. MEC units aim to achieve high aggregate goodput, stable multi-tenant isolation, efficient site-level power usage, and sustained server utilization during peak demand [32,33]. Microgrid μ DCs optimize the relation between energy and computation, sustain autonomous operation under renewable supply, and preserve service stability under significant power variability [36,38]. Underwater μ DCs operate within sealed thermal envelopes with liquid cooling as the primary mechanism, maintaining high service availability, and exhibit end-to-end performance shaped by the subsea path length and protocol overheads over fiber links [13,15]. Space μ DCs optimize for long-duration autonomous operation and radiation-tolerant fault handling with timely recovery, emphasize task-level energy efficiency on specialized processors, and employ bandwidth-efficient offloading through aggressive compression under downlink constraints [16].

Motivated by these typology-specific requirements, this survey consolidates performance metrics across existing studies and, by synthesizing established reporting practices, develops a compre-

hensive attribute-oriented framework for comparing edge and micro datacenters. Reported metrics consistently organize into six attributes: 1) *Service quality* focuses on tail latency behavior, stability at high percentiles, and compliance with stated service level objectives. 2) *Resource efficiency* reflects useful work per unit energy and effective utilization across compute, memory, and network. 3) *Sustainability* encompasses power usage effectiveness, thermal design margins, and capacity to integrate renewable energy at the site. 4) *Reliability and resilience* are assessed through failure characteristics, recovery capability, and continuity of service during faults and upgrades. 5) *Adaptability and scalability* consider autoscaling responsiveness, sensitivity to interference, and elasticity under fluctuating demand and changing topology. 6) *Economic efficiency* evaluates cost per unit work and cost per successful outcome. Although individual studies instantiate these attributes with heterogeneous measures and system boundaries, the framework provides a unified taxonomy for comparative analysis aligned with the priorities above.

To operationalize this framework, performance characteristics manifest hierarchically across three operational levels. At the infrastructure level, choices in accelerators, memory hierarchies, network topologies, power delivery, and cooling establish fundamental limits on throughput, latency, energy draw, and safe operating temperatures, thereby defining the achievable envelope before software policies are applied. At the platform level, container orchestration, topology-aware scheduling, quality-differentiated networking, reactive autoscaling, and non-disruptive update mechanisms shape tail latency distributions, service attainment, multi-tenant utilization, and fairness, translating raw capacity into delivered performance under re-

alistic contention. At the application level, workload semantics determine which attributes are most dominant. Latency-sensitive control loops emphasize tight tail stability, low jitter, and strong session continuity, whereas batch analytics focuses on sustained throughput, high accelerator utilization, and energy proportionality across the load range.

Because micro datacenters operate under edge-specific constraints, conventional metrics must be recalibrated to remain decision-useful. Mobility introduces strict budgets for handover events, bounded state migration overheads, and strong requirements for session preservation during context transition. Energy and thermal considerations extend beyond campus-level reporting to site-level battery autonomy, local renewable contribution, and passive cooling capability at a given volumetric density, with attention to safe thermal margins within enclosure constraints. Reliability must reflect limited on-site maintenance, which elevates the importance of rapid remote recovery, controlled upgrade risk, and repair actions that are feasible under constrained physical access. Reduced scale motivates micro PUE reported at rack or pod granularity, emphasizing local efficiency rather than facility-level aggregates and enabling finer attribution of power and cooling improvements. Together, these adjustments link practical priorities to a stable comparison vocabulary and to concrete levers across system levels.

3 μ DC Infrastructure

3.1 Performance Metrics

This section highlights the core infrastructure enabling μ DCs, focusing on compute, networking, structure, power, and cooling. Table 3 summarizes representative deployments carefully se-

lected to reflect diversity across use cases and technical challenges. Vehicular μ DCs demonstrate mobile, automotive-grade integration [1, 41, 42]. IIoT μ DCs illustrate ruggedized, containerized solutions for industrial edge environments [10, 43, 44]. MEC units showcase telecom-integrated, low-latency platforms [45–47]. Microgrid μ DCs are chosen for their emphasis on sustainable power and flexible deployment [36, 40, 48]. Underwater and space-based μ DCs represent advances in resilient and remote-edge computing [13, 16, 20, 49]. These examples together capture the breadth of μ DC design strategies and the key considerations that inform their infrastructure choices.

3.2 Compute and Storage

The computational foundation of an μ DC determines performance, latency, energy efficiency, and the capacity to support heterogeneous workloads. Common compute architectures include server clusters, SoC clusters, and wafer-scale units, each with distinct tradeoffs in performance, power, and integration density. Across deployments, these configurations aim to balance workload demand against resource limits. Platform choice shapes the outcome: server-based nodes improve multitenant fairness despite virtualization overhead; SoC clusters reduce cold start penalties and stabilize tail latency under tight power budgets; wafer-scale units deliver maximal throughput density at narrower thermal margins.

Compute Unit. Most μ DCs adopt clustered server architectures built on processors with virtualization support, offering flexibility and broad compatibility across MEC and subsea deployments [46, 50]. Lightweight SoC-based designs, prevalent in vehicular and IIoT settings, deliver superior energy efficiency and compact footprints [25, 42,

Table 3 Detailed Infrastructure Features and Representative Cases of μ DC Deployments.

μ DC Typologies	Typical Cases	Compute/Storage			Network			Structural Cfg.		Energy		Cooling	
		Compute Unit (<u>Server</u> / <u>SoC</u> / <u>Wafer</u>)	Heterogeneity (<u>GPU</u> / <u>GPU</u> / <u>NPU</u> / <u>FPGA</u>)	Storage Type (<u>Local</u> / <u>Shared</u>)	Connectivity Type (<u>Wired</u> / <u>Wireless</u>)	Bandwidth Capacity (<u>High</u> / <u>Medium</u> / <u>Low</u>)	Latency Expectation (<u>High</u> / <u>Medium</u> / <u>Low</u>)	Structure Factor (<u>Container</u> / <u>Rack</u> / <u>Modular</u>)	Scalability (<u>Scale-Out</u> / <u>Scale-Up</u>)	Energy Source (<u>Grid</u> / <u>Renewable</u> / <u>Hybrid</u>)	Energy Backup (<u>Battery</u> / <u>UPS</u> / <u>Generator</u>)	Cooling Type (<u>Passive</u> / <u>Active</u>)	Cooling Source (<u>Air</u> / <u>Liquid</u> / <u>Space</u>)
Vehicular μ DC	<i>Nvidia DRIVE Thor</i> [1]	So.	C	S	Wls	M	L	Mo.	SO	H	B	A	Air
	<i>Serverless Vehicular</i> [41]	Ser.	C-G	L	Wls	M	L	R	SU	G	U/Ge	A	Air
	<i>Mobileye</i> [42]	So.	C-N	L	Wls	H	L	Mo.	SO	H	B	A	Air
IIoT μ DC	<i>FogROS</i> [10]	Ser.	C-N	S	Wd	H	L	Con.	SO	G	U	A	Air
	<i>Schneider R-Series</i> [43]	Ser.	C-G	L	Wd	H	L	Con.	SU	G	U	A	Air
	<i>Dell VxRail OT Edge</i> [44]	Ser.	C-N	S	Wd	H	L	R	SO	G	U	A	Air
MEC Unit	<i>Wafer-Scale Computing</i> [45]	W	C-G	S	Wd	H	L	Mo.	SU	G	U/Ge	A	Air
	<i>AWS MEC</i> [46]	Ser.	C	S	Wls	H	L	R	SO	G	U/Ge	A	Air
	<i>Verizon 5G Edge</i> [47]	Ser.	C-N	S	Wls	H	L	R	SO	G	U/Ge	A	Air
Microgrid μ DC	<i>Liquidstar Waypoint</i> [48]	So.	C	L	Wls	M	M	Con.	SO	R	B	P	Liq.
	<i>Urban μDC</i> [36]	Ser.	C-G	L	Wd	M	M	Mo.	SO	H	B	A	Air
	<i>Junkyard Computing</i> [40]	Ser.	C	L	Wls	M	M	Con.	SO	H	B	A	Air
Underwater μ DC	<i>Project Natick</i> [13]	Ser.	C	L	Wd	H	L	Con.	SU	R	-	A	Liq.
	<i>Subsea Cloud Pod</i> [49]	Ser.	C	L	Wd	H	L	Con.	SO	G	B	P	Liq.
Space μ DC	<i>Tiansuan Constellation</i> [20]	W	C-N	L	Wls	L	H	Mo.	SU	R	B	P	Sp.
	<i>Space Microdatacenter</i> [16]	So.	C-G	L	Wls	L	H	Mo.	SU	R	B	P	Sp.

[48, 51]. A small but growing segment now leverages wafer-scale systems to sustain high-density AI workloads [24, 45]. In practice, stronger isolation improves SLO adherence under contention, whereas lean software stacks curb orchestration overheads that often dominate tail latency.

Heterogeneity. μ DC platforms integrate heterogeneous compute resources, including GPUs, ASICs, and FPGAs, to support latency-sensitive applications such as perception, robotics, and industrial control. This architectural diversity often requires hardware-aware scheduling and accelerator-specific service partitioning. Hardware-aware partitioning raises goodput at fixed power budgets and bounds interference across accelerators.

Storage Type. In terms of storage, server-class μ DCs typically employ shared or distributed storage solutions to facilitate service migration,

caching, and data replication across nodes [46, 47, 49]. SoC-based and off-grid μ DCs generally use local, isolated storage for simplicity and operational autonomy [25, 48]. This trade-off manifests in failover time and energy efficiency—shared storage accelerates migration but may exacerbate access tails, whereas local storage favors autonomy at the expense of longer recovery latencies.

3.3 Network Connectivity

Robust, context-aware networking is crucial for μ DCs, especially in distributed, mobile, or remote environments. Network infrastructure shapes bandwidth and latency, and also governs integration across edge-to-cloud and edge-to-edge topologies. Key performance drivers include link asymmetry, which raises tail latency; queueing behavior, which sets jitter bounds; and mobility support, which determines handover continuity.

Connectivity Type. μ DCs employ various communication models that combine both wired and wireless interfaces. Wired connections such as Ethernet and optical fiber are common in fixed-location deployments, particularly in MEC systems and subsea infrastructures [13, 43]. Wireless communication, including 5G, V2X, Wi-Fi, and LoRa, is frequently used in vehicular and IIoT μ DCs to support mobility and deployment flexibility [1, 25, 42, 48].

Bandwidth Capacity. Serve and telecom-grade μ DCs typically operate under high-bandwidth environments, especially those supported by fiber or 5G backhaul [46, 47]. Medium-bandwidth configurations are found in Wi-Fi-based IIoT deployments [25, 51], while μ DCs in rural or solar-powered settings often operate with limited bandwidth through satellite or LPWAN links [48, 49].

Latency Expectation. The latency requirement of μ DCs varies by workload type. Vehicular and MEC-based μ DCs prioritize low-latency response, often under 10ms [1, 6, 42], IIoT and edge robotic systems operate within medium-latency ranges [25, 51], whereas high-latency tolerance is acceptable in remote, underwater, or batch-oriented deployments [49, 50] where real-time responsiveness is not critical.

3.4 Structural Configuration

The physical structure of μ DC's infrastructure reflects its environmental constraints, deployment modality, and maintenance requirements. Structure directly impacts thermal headroom, airflow efficiency, and service accessibility—jointly determining availability and recovery time.

Structure Factor. In most cases, μ DCs adopt one of three form factors: rack-mounted enclosures, containerized systems, or modularized edge

units. Rack-based installations dominate conventional deployments in controlled industrial, telecom, and edge-cloud environments. Systems such as the Schneider R-Series and the Dell VxRail OT Edge utilize standard 19-inch racks or wall-mounted enclosures to ensure compatibility with existing IT infrastructure [43]. Containerized μ DCs are favored in harsh environments or mobile use cases. Project Natick and Subsea Cloud Pod deploy sealed containers underwater [49, 50], whereas off-grid deployments, such as Liquidstar Waypoint, utilize lightweight pods designed for solar-powered microgrids [48]. Modular units emphasize field-assembled scalability and hot-swappable nodes, often tailored for telecom or rural applications [46].

Scalability. From a scalability perspective, most μ DCs favor scale-out architectures, in which additional units or modules are deployed laterally to increase the system's computing capacity. This applies to telecom-grade MEC systems and microgrid nodes alike. In contrast, a few systems rely on scale-up designs, vertically enhancing compute density within fixed enclosures, such as wafer-scale systems or space-constrained vehicular μ DCs [1, 7]. Streamlined designs reduce mean time to repair, while scale-out architectures improve elasticity with coordination overhead, and scale-up approaches concentrate thermal density, requiring careful margin management.

3.5 Power and Energy

Power supply is foundational to modern micro datacenter design. Scarcity of power resources presents the primary bottleneck, constraining scalability, high-performance computing support, and deployment feasibility. Crucially, power architecture choices, from grid dependency to modular re-

newables and distributed storage, directly determine μ DC cost-effectiveness, expansion capacity, carbon footprint, and environmental sustainability [52]. A robust power system is therefore essential for enabling incremental growth, integrating green energy, and ensuring resilience during disruptions. Innovations such as hybrid power provisioning and intelligent management are crucial for achieving a balance between reliability, efficiency, and environmental sustainability goals [53]. Key metrics span autonomy duration, ride-through capability during undervoltage events, switchover recovery time, and micro-PUE at rack scale.

Energy Source. μ DCs typically adopt one of three energy sourcing strategies: grid-connected, renewable-powered, or hybrid configurations. Grid-powered μ DCs are common in industrial and telecom deployments, leveraging stable utility access to ensure consistent operation. Notable examples include the Schneider R-Series, AWS MEC, and EdgeX Foundry, all of which draw electricity from municipal or enterprise grids [43, 46, 51]. In contrast, renewable-powered μ DCs are designed for off-grid or sustainability-driven scenarios; for instance, Liquidstar Waypoint operates entirely on solar energy for both compute and storage functions [48]. Microgrid solutions, such as the RISE Smart Microgrid, integrate solar arrays with real-time energy management systems. Hybrid models combine solar generation, battery storage, and backup grid or diesel input to maximize reliability and operational resilience. These choices affect tail stability under supply fluctuations and determine leveled energy costs.

Energy Backup. To ensure operational continuity during power outages, most micro datacenters incorporate battery-based or UPS-backed reserves for short-term backup [54]. Traditional UPS sys-

tems are standard in grid-connected installations, while battery-only solutions are favored in mobile or embedded μ DCs, such as those from Mobileye and Liquidstar [42, 48]. For extended backup in remote or demanding environments, diesel generators are often deployed to provide sustained power availability. Critical measures include ride-through duration and recharge time to restore SLO capacity.

3.6 Cooling System

Cooling system design in μ DCs requires careful optimization of thermal efficiency within structural and environmental constraints. Unlike traditional datacenters, which employ complex, centralized cooling infrastructures, μ DCs integrate thermal management directly within each modular container unit as an essential architectural component [53]. This integration fundamentally defines operational boundaries: cooling effectiveness establishes safe operating margins, determines hotspot mitigation rates, and influences long-term reliability through thermal stress cycles.

Cooling Type. μ DCs implement either passive or active cooling strategies, depending on deployment requirements. Passive cooling predominates in lightweight, low-power deployments such as Liquidstar Waypoint and EdgeX Foundry, which eliminate mechanical components and utilize convection and radiation for heat dissipation [48, 51]. This approach is ideal for solar-powered, fanless, and hermetically sealed environments, where energy efficiency and silent operation remain paramount. Conversely, server-class μ DCs processing substantial thermal loads employ active cooling through fans, blowers, or dedicated thermal loops, as demonstrated by NVIDIA DRIVE Thor, AWS MEC, and Project Natick [1, 13, 46]. Active strategies stabilize p99 latency during load

bursts while incurring additional power overhead, whereas passive approaches maximize energy efficiency within stricter thermal envelopes.

Cooling Source. Air cooling remains the primary choice for terrestrial μ DCs owing to operational simplicity and economic viability. However, liquid cooling increasingly serves high-density deployments, including Subsea Cloud Pod [49] and wafer-scale architectures [7,45,55,56], where computational density necessitates superior heat extraction capabilities. Select μ DCs exploit environmental thermal gradients directly; underwater deployments, such as Project Natick, utilize seawater cooling [13], while future orbital platforms may leverage the space environment’s natural thermal dissipation properties [16]. Critical performance indicators encompass thermal violation rate, inlet temperature sensitivity, and enclosure micro-PUE.

Recent micro and edge datacenter cooling strategies center on three key directions. First, immersion and sub-ambient cooling precisely regulate server temperatures, optimizing power efficiency and reducing total cost of ownership without hardware changes [57]. Second, modular and containerized designs with adaptive cooling, such as in-row or overhead air delivery, minimize airflow inefficiencies and support workload consolidation, yielding substantial energy savings [58]. Third, fine-grained warm water cooling dynamically adjusts temperature and flow at the hardware level, while advanced cold plates with vapor chambers rapidly relieve hotspots, enabling efficient cooling in high-density, heterogeneous environments [59]. Together, these approaches exemplify the shift toward workload-aware and energy-efficient cooling in modern distributed datacenters.

4 μ DC Platform

This section analyzes representative μ DC platform technologies across several key categories, as summarized in Table 4. Hardware–software co-designed platforms [5, 25, 56, 60–66] demonstrate integrated approaches to balancing performance, power, and scalability. Purely software-centric solutions [6, 13, 21, 40] focus on orchestration, control, and operational flexibility. System-level and architectural innovations [7, 16, 20, 23, 24, 55, 67] highlight advances in resource management and multi-tenancy. These categories, and the selected works within them, collectively illustrate the diversity of μ DC platform designs and the evolving requirements of edge deployments. Platform choices influence tail behavior and SLO compliance. Employ a hybrid approach, combining custom hardware with advanced software to strike a balance between spectrum from hardware-centric [13, 24, 45] to software-centric solutions [6, 25, 68]. Hardware-driven platforms emphasize compute density, power efficiency, and tight integration, often leveraging specialized SoC clusters or wafer-scale chips, such as Tesla Dojo [7], Cerebras CS-2 [56], and the UCLA&UIUC Waferscale-Chip [55], to optimize performance, energy, and cost tradeoffs. In contrast, software-centric systems prioritize flexibility and compatibility with cloud native and middleware ecosystems, enabling distributed orchestration, containerization, and multi-tenancy. Some solutions [5,20,24] employ a hybrid approach, combining custom hardware with advanced software to strike a balance between performance, energy efficiency, and deployment agility for embedded or edge applications. From a performance standpoint, hardware-centric stacks tend to compress tail latency and improve energy effi-

Table 4 Platform-Level Design Patterns, Parameter Settings, and Resource Management Techniques in μ DCs.

Platform	Institution	Type (HW/SW)	Overhead (Perf/Power/ Cost)	Parameter Setting (Static/Dynamic/ Hybrid)	Resource Mgmt. (Non/VM/Container)	Data Flow (Aggregation/Sharing/ Offloading)	Control Pattern (Distributed/Centralized/ Hierarchical)	Multi-Tenancy (STSU/MTSU/ STMU/MTMU)
A2 System [5]	Academic	HW+SW	Perf	H	Non	O	C	STSU
ASADI [60]	Academic	HW+SW	Perf+Power	H	Non	A	Hi	STSU
Cerebras CS-2 [56]	Industry	HW	Cost	H	Non	O	D	MTMU
CLOUDS-Pi [61]	Academic	HW+SW	Cost+Perf	D	VM	Sh	C	MTMU
DOME MicroDC [62]	Industry	HW	Cost	S	Non	A+O	Hi	STSU
FogROS 2 [25]	Industry	SW	Perf	H	VM/Container	O	C	MTMU
FORMS [63]	Academic	HW	Perf+Cost	H	Non	A	Hi	STSU
In-Situ AI [64]	Academic	HW+SW	Perf+Power+Cost	D	VM/Container	A+O	Hi	MTMU
InS-RA [65]	Academic	HW+SW	Perf+Power	D	VM/Container	A	Hi	MTMU
InSURE [66]	Academic	HW+SW	Perf+Power+Cost	H	VM/Container	A+O	Hi	MTMU
Jigsaw System [6]	Academic	SW	Perf	H	Non	A+Sh	C	MTMU
Junkyard Computing [40]	Academic	HW+SW	Perf+Power+Cost	D	Container	Sh+O	Hi	MTMU
Project Natick [13]	Industry	HW	Cost	S	Non	O	C	MTMU
SatelliteFL [21]	Academic	HW+SW	Power	H	Non	A	C	STSU
SoC-Cluster [23]	Academic	HW+SW	Perf	H	Container	A+O	D	MTMU
SoCFlow [24]	Academic	HW+SW	Perf+Power	H	Non	A+O	Hi	MTMU
Space μ DC [16]	Academic	HW+SW	Power	H	Container	O	Hi	MTMU
StarlingX [67]	Industry	SW	Perf	H	Container	A+O	Hi	MTMU
Tesla Dojo Chip [7]	Industry	HW	Cost	S	Non	Sh	C	STSU
Tiansuan Constellation [20]	Academic	HW	Power	H	Container	O	Hi	MTMU
UCLA&UIUC Chip [55]	Academic	HW	Power+Cost	H	Non	A	Hi	STMU

ciency at high load, whereas software-centric approaches expand adaptability at a measurable orchestration cost.

The design overhead of μ DC platforms varies according to system objectives, including performance, power consumption, cost, or combinations thereof. High-performance and energy-efficient designs [5, 24, 40] are tailored for latency-sensitive and resource-constrained edge scenarios. Wafer-scale architectures, such as Tesla Dojo [7], target cost-effective scaling, while the Cerebras CS-2 optimizes for high-throughput AI inference [45, 56]. Software-oriented platforms, such as StarlingX [67] and FogROS 2 [25], focus on platform resilience, orchestration reliability, and elasticity. Notably, platforms like Junkyard Computing prioritize comprehensive efficiency under resource-constrained and cost-sensitive conditions, reflecting practical deployment concerns. Critical measurements include cold start latency, context-switch overhead, convergence time, and amortized cost per task.

μ DC platforms employ diverse parameter-setting strategies, ranging from static [13, 62] to dynamic [25, 61, 67] and hybrid configurations [24, 40, 45]. Dynamic and hybrid strategies enable real-time policy adaptation, including resource allocation, energy management, and service prioritization, thereby enhancing elasticity and robustness in unpredictable edge environments. Static configurations are typically found in isolated, long-duration deployments where system adaptability requirements are minimal. Dynamic configurations exhibit reconfiguration latency and potential oscillation that must be balanced against improved steady-state efficiency.

Resource management in modern μ DCs predominantly utilizes container-based orchestration, enabling lightweight deployment, modular scaling, and improved fault isolation [16, 20, 23, 40]. Several platforms support both virtual machine and container paradigms [25], facilitating seamless integration of hybrid edge-cloud environments. Non-virtualized [5, 21, 24, 45, 62] resource management

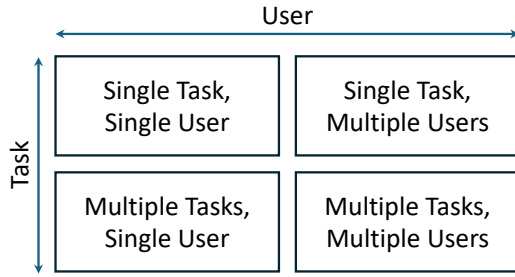


Fig. 2 A taxonomy of tenancy-based architectures for resource management of μ DCs

is generally reserved for highly specialized or ultra-low-power designs, minimizing system overhead for maximum efficiency.

Data flow models in μ DC platforms include aggregation, sharing, and offloading, tailored to diverse application requirements. Control patterns range from centralized and distributed to hierarchical. Aggregation and offloading are crucial for edge-cloud collaboration, enabling low-latency inference and batch analytics, while distributed or hierarchical control schemes enhance network autonomy, resilience, and scalability.

Following recent proposals for resource management architectures that often feature finely divided hierarchies or functionalities, and building on classifications based on the number of physical and logical layers [69], we introduce a new taxonomy for μ DC platforms centered on task and user characteristics. The task dimension refers to the number and nature of computational workloads concurrently managed on the platform, which impacts the granularity of resource allocation. The user dimension defines the user concurrency model and associated access policies, directly influencing interaction patterns and isolation requirements. Based on these axes, multi-tenancy in μ DCs can be classified by the number of tasks supported on a given platform, resulting in four fundamental models:

- *Single Task, Single User (STSU)*: The μ DC

platform executes a single task for a single user. Both the task and the user have exclusive access to the hardware, which is typical for dedicated deployments.

- *Single Task, Multiple Users (STMU)*: The μ DC runs one task that serves multiple users. While the application remains exclusive, the system accommodates multiple simultaneous user requests.
- *Multiple Tasks, Single User (MTSU)*: The platform supports several distinct tasks, but each is dedicated to a single user, a configuration suitable for experimental or development use cases.
- *Multiple Tasks, Multiple Users (MTMU)*: The most flexible model, in which multiple tasks are hosted and can be accessed by multiple users, representing an ideal public edge or μ DC infrastructure with maximized resource sharing and efficiency.

As shown in Figure 2, the model progresses toward *MTMU*, both the diversity of tasks and the breadth of user access increase, supporting greater resource sharing and system utilization. Some platforms [20, 24, 34, 40, 45, 67] that support multiple tasks and users are designed to enable efficient resource pooling, robust isolation, and elastic scheduling, making them well-suited for complex, multi-tenant edge environments. In contrast, platforms [21, 45, 62] optimized for single-task deployments emphasize performance, determinism, or operational longevity, targeting mission-specific or dedicated workloads. This tenancy-based taxonomy provides a clear framework for analyzing μ DC platform capabilities, guiding both architectural design and resource management strategies across diverse scenarios.

Table 5 Optimization Objectives, Techniques, and Limitations for μ DC Application Scenarios.

Works	Application (BE/LC)	Optimal Objective	Dimension (Temporal/ Spatial)	Scope (Global/ Local)	Granularity (System/Service/ Task/Data/Model)	Technique	Limitation
Aether [70]	LC	Performance	T	G	Service	Dynamic scheduler	QoS violation
AWS Snowball Edge [71]	LC	Power/Energy	T	L	Model	Quantization	Hardware-specific
CLOUS-Pi [61]	LC	Security	S	L	Data	Encrypted alert	Key management
DOME MDC [62]	BE	Reliability	S	G	System	Edge orchestration	Coordination overhead
EdgeFabric [72]	BE	Reliability	T	G	Service	Decentralized SGD	Staleness
EdgeX Foundry [51]	LC	Power/Energy	T	L	Task	Adaptive sensing	Sampling loss
FogROS 2 [25]	LC	Accuracy	T	L	Task	Stream windowing	Sync overhead
InSURE [66]	BE	Sustainability	T+S	G	System	Energy buffering	Intermittent supply
Intel Mobileye [42]	LC	Reliability	S	L	Service	Failover routing	Delay overhead
K3S Surveillance [73]	LC+BE	Reliability	S	G	Task	Multi-tier task assignment	Preemption risk
LiquidStar [48]	BE	Power/Energy	T	L	Service	Microservice fusion	Service diversity
NEMO [74]	LC+BE	Utilization	T+S	L	Task/System	Slack, Pre-warm	Fragmentation
NVIDIA Drive AGX [75]	LC	Perf	T	L	Data	Sensor fusion	Sensor inconsistency
OCS [76]	LC+BE	Delay	T+S	G	Task	Online scheduling	Uncertainty
OLC [77]	LC+BE	Cost, QoS	T+S	G	Task	Lazy caching	Cold start
OpenNESS AI [78]	LC+BE	Performance	T	G	Service	Pipeline isolation	QoS interference
PowerSync [79]	LC+BE	Efficiency	T+S	G	Task	Power sync and Profiling	Sync delay
RRZP-Zygote [80]	LC+BE	Startup time	T+S	L	Task	Zygote planning	Memory Exclusivity
SoCFlow [24]	BE	Utilization	S	G	Model	Compression	Latency sensitivity
SSAP-MEC [81]	LC+BE	Cost, Delay	T+S	G	Task	State-aware placement	Uncertainty
StarlingX [67]	BE	Reliability	S	G	Task	Multi-tenant model	Complex policy
Wafer-Scale Computing [45]	LC	Perf	T	L	Model	Wafer-scale acceleration	Heat density

5 μ DC Application

To provide a holistic view of application optimization in μ DCs, Table 5 presents a comparative overview of representative works, systematically categorizing the spectrum of μ DC workloads, optimization objectives, and technical approaches. These cases are carefully selected to capture the diversity of real-world deployment scenarios and the evolution of strategies for balancing performance, reliability, energy efficiency, and system scalability. Solutions are organized according to key dimensions, including temporal and spatial optimization, local versus global management scope, and varying levels of granularity from system orchestration to fine-grained data and model optimization. This structured perspective enables direct comparison across different μ DC paradigms.

Latency-critical applications represent a major focus within μ DC deployments. Typical scenarios include autonomous driving, real-time perception, and mission-critical sensor fusion, where sys-

tems must deliver ultra-low latency and high reliability [25, 42, 45, 51, 70–74]. To satisfy these requirements, solutions frequently incorporate techniques such as dynamic scheduling, adaptive sensing, and hardware acceleration, ensuring that data is processed and responded to within strict temporal constraints. These approaches are essential in safety-critical and interactive environments, where any delay may directly impact system functionality or user experience.

Best-effort workloads constitute another important category, encompassing distributed analytics, system orchestration, and sustainable computing tasks deployed in large-scale or resource-constrained environments [13, 24, 48, 62, 67, 75]. For these scenarios, optimization strategies typically emphasize reliability, energy efficiency, and coordination across diverse edge resources. Techniques such as edge orchestration, resource compression, and decentralized management frameworks are widely adopted to improve overall system utilization, maintain service availability, and

adapt to fluctuating workload demands in harsh or dynamic operating conditions.

In practice, many μ DC frameworks are required to accommodate both LC and BE tasks within a unified platform. These hybrid or multi-tenant environments demand flexible solutions capable of adaptive scheduling, efficient resource sharing, and state-aware placement [61, 66, 76–81]. Such systems highlight the need for multi-objective optimization, balancing competing demands for latency, throughput, energy, and reliability across heterogeneous workloads and deployment contexts. The increasing prevalence of these hybrid workloads underscores the evolution of μ DCs toward more versatile computing infrastructures.

Across all categories, μ DC optimization techniques can be broadly classified into temporal and spatial domains, each addressing distinct operational challenges. Temporal optimization focuses on reducing response time and synchronizing processing intervals to meet the demands of latency-sensitive applications. Common temporal strategies include deadline-aware scheduling, adaptive windowing, quantization, and real-time stream processing, as exemplified by works such as [25, 45, 51, 71, 72]. These techniques are typically employed in scenarios requiring stringent timing guarantees, such as real-time sensor fusion, dynamic task assignment, or adaptive sensing in mission-critical environments.

Spatial optimization focuses on efficiently placing workloads and allocating resources across multiple μ DC nodes or distributed clusters. Representative spatial approaches include topology-aware placement, federated coordination, workload colocation, and multi-tier assignment, as demonstrated in studies like [13, 24, 62, 67, 73]. These methods aim to maximize resource utilization, improve sys-

tem reliability, and enhance scalability by dynamically distributing tasks based on physical topology, network state, or geographic proximity.

Notably, several optimization techniques span both temporal and spatial dimensions, integrating real-time responsiveness with distributed resource management. For instance, adaptive scheduling and online task orchestration [61, 77, 81] can dynamically allocate workloads in response to both system state and application deadlines. Other approaches, such as microservice fusion [48] and state-aware placement [78], simultaneously consider timing constraints and spatial distribution to optimize overall efficiency and resilience. This dual-domain optimization is increasingly vital as μ DCs evolve to support complex, heterogeneous workloads across diverse deployment scenarios.

The granularity of these optimization strategies ranges from system-wide orchestration and service-level adaptation to fine-grained task, data, and model management, as reflected in Table 5. This multi-level approach allows μ DCs to flexibly balance trade-offs among performance, efficiency, reliability, and scalability. In summary, the surveyed works reveal that effective μ DC optimization increasingly relies on the joint consideration of temporal, spatial, and granular dimensions, enabling robust and adaptive resource management for diverse edge computing scenarios. The surveyed works collectively capture the state of the art in μ DC application optimization, revealing not only diverse technical pathways but also common design trade-offs and limitations. This comparative analysis provides a foundation for identifying future opportunities and emerging challenges in the deployment of high-performance, resilient, and energy-efficient μ DCs.

Table 6 Challenge–Opportunity–Technique matrix across four layers. μ DC Types use numeric markers: ① Vehicular μ DC; ② IIoT μ DC; ③ MEC Unit; ④ Microgrid; ⑤ Underwater μ DC; ⑥ Space μ DC.

Layer	Challenge	Opportunity	Key Metrics	MDC Types	Possible Techniques
Infrastructure	Intermittent Power	Hybrid Energy Mgmt	Autonomy Hrs; mPUE	①④⑤	Solar/Wind; Supercap Buffer; Power Capping
	Extreme Temperature	Adaptive Cooling	Op Range; Thermal Eff	①⑤⑥	PCM; Liquid Loop; Radiative Cooling
	Vibration & Shock	Mech Resilience	Shock (g); MTBF	①③⑤⑥	Active Damping; Rugged SSDs; Shock Mounts
	Pressure Variation	Env Sealing	Depth/Altitude; IP69K	⑤⑥	Pressure Vessel; Hermetic Seal; Pressure Equalization
	Limited Volume	Ultra-Dense Compute	TFLOPS/L; Mass Budget	①③⑥	3D Stacking; Immersion; Modular Chassis
	Corrosion Risk	Material Protection	Corrosion Rate; Salt-Fog Hrs	③⑤	Conformal Coat; Ti Enclosure; Cathodic Protect
Platform	Network Partitioning	Autonomous Ops	Offline Duration; Consistency	①⑤⑥	Split-Brain; Local FSM; Eventual Consistency
	Resource Heterogeneity	Unified Orch	Sched Eff; Utilization	②③④	Container Runtime; HAL; Resource Pooling
	Remote Management	ZTP	MTTR; Automation Cov	④⑤⑥	Satlink Mgmt; OOB Access; Self-Heal Policies
	Multi-Tenancy	WL Isolation	Perf Isolation; QoS	②③④	SR-IOV; Kata; Time-Partitioned
	Firmware Updates	Secure OTA	Update Success; Rollback Time	①④⑥	A/B Partitions; Secure Boot; Delta Updates
	Trust Establishment	HW RoT	Attest Latency; TPM Coverage	②③⑥	TPM 2.0; Secure Enclave; Remote Attestation
Application	Motion Compensation	Dynamic Stabilization	Position Error; Comp Latency	①⑤⑥	IMU Fusion; Kalman; Predictive Comp
	Bandwidth Scarcity	Edge AI	Local Rate; Reduction Ratio	①⑤⑥	FL; Edge Infer; Semantic Comp
	Real-Time Constraints	Deterministic Exec	WC Latency; Jitter	①②③	Time-Triggered; RTOS; HW Timers
	Safety Criticality	Fail-Safe Ops	SIL; Safe-State Time	②④⑥	TMR; Watchdog; Formal Verif
	Data Sovereignty	Local Processing	Residency; Compliance Rate	②③④	HE; SMPC; TEE
	Model Deployment	Adaptive Models	Accuracy; Infer Latency	②③④	Quantization; NAS; TinyML
Cross-layer	Power–Thermal Coupling	Holistic Energy	Perf/W; Headroom	①④⑥	DVFS; Workload Shaping; Predictive Cooling
	End-to-End Reliability	Mission Assurance	Avail (9s); MTBSI	②⑤⑥	BFT; C/R; Diversity
	Observability Gaps	Distrib Diagnostics	Telemetry Cov; Detect Rate	①③⑤	Tracing; Black-Box Logging; Synthetic Mon
	Resource Scarcity	X-Layer Optimization	Resource Eff; SLA Hit Rate	①⑤⑥	Hierarchical Control; Game-Theoretic; ML Orchestration

Abbr.: mPUE = micro Power Usage Effectiveness; TFLOPS/L = teraFLOPS per liter; MTBSI = mean time between service incidents; RoT = (hardware) root of trust; ZTP = zero-touch provisioning; OOB = out-of-band; C/R = checkpoint/restart.

6 Challenges and Opportunities

The ongoing evolution of μ DCs presents both compelling possibilities and formidable challenges that span infrastructure, platform, application, and cross-layer integration. As μ DC deployments become increasingly diversified and mission-critical, the research community faces the imperative to reimagine both the architectural foundations and operational paradigms that will underpin the next generation of edge-centric data infrastructures. To operationalize the discussion, we compile a concise, layer-organized table that maps challenges to opportunities and design directions, with cross-layer couplings explicitly marked in Table 6.

6.1 Design Challenges

This subsection outlines the principal design challenges of micro datacenters across four layers, in-

cluding infrastructure, platform, application, and cross-layer. It summarizes environment-driven constraints, control plane complexities, workload requirements, and system-wide couplings that together shape risk profiles and performance envelopes, providing the basis for the subsequent mapping to opportunities and design directions.

Infrastructure Layer. Micro datacenters face foundational constraints that shape deployment viability: 1) Intermittent power in vehicular, microgrid, and underwater settings elevates autonomy and PUE concerns; 2) Extreme thermal conditions in vehicular, underwater, and space environments strain reliability under limited cooling capacity; 3) Mechanical stress from vibration and shock accelerates wear across vehicular, edge, underwater, and orbital installations; 4) Pressure differentials and corrosive atmospheres in underwater, space, and harsh edge sites require sealed, resilient enclosures

and robust material choices; 5) Volumetric constraints on vehicles, edge sites, and spacecraft limit compute density and tighten the trade off between capacity and size.

Platform Layer. Control plane operations in distributed edge settings encounter distinct challenges: 1) Network partitioning over intermittent, high latency links in vehicular, underwater, and space contexts demands coordination and consistency under isolation; 2) Heterogeneous resources across industrial IoT and edge sites complicate abstraction and scheduling; 3) Constrained remote management in microgrid, underwater, and space environments increases the need for automated remediation and predictive maintenance; 4) Multi tenant sharing requires strong performance isolation to prevent SLO violations; 5) Secure firmware provisioning with low overhead cryptographic attestation is essential across vehicular and industrial IoT systems.

Application Layer. Edge applications impose specific operational demands: 1) Platform motion on vehicular, underwater, and space systems introduces estimation drift that requires continuous compensation; 2) Bandwidth limits drive intelligent data reduction and push computation closer to sources to reduce transmission overhead; 3) Deterministic timing in vehicular, industrial IoT, and edge workloads prioritizes temporal predictability over average performance; 4) Safety critical operations in industrial IoT, microgrid, and space require verifiable failsafe mechanisms and standards compliant certification; 5) Data residency must be reconciled with adaptive model deployment in industrial IoT, edge, and microgrid settings to balance compliance, model evolution, and inference efficiency.

Cross-Layer. System wide effects cut across lay-

ers and demand holistic control: 1) Power and thermal coupling in vehicular, microgrid, and space tightly links energy supply, heat dissipation, and compute performance; 2) Mission critical reliability in industrial IoT, underwater, and space calls for coordinated redundancy to sustain availability; 3) Limited observability in vehicular, edge, and underwater settings necessitates anomaly detection over compressed or sampled telemetry; 4) Pervasive resource scarcity on vehicular, underwater, and space platforms requires adaptive allocation of compute, storage, and network as conditions change.

6.2 Future Opportunities

This subsection identifies forward-looking opportunities aligned with the Infrastructure, Platform, Application, and Cross-Layer perspectives. For each layer, we identify concrete design directions that address the preceding challenges and highlight leverage points for energy autonomy, thermal and mechanical resilience, trustworthy control, verifiable safety, and efficient adaptation. These opportunities align with the entries in Table 6 and highlight where cross-layer couplings can enhance system benefits.

Infrastructure Layer. Architectural innovations mitigate infrastructure limits across vehicular, microgrid, underwater, edge, and space deployments: 1) Hybrid energy architectures blend renewables, storage buffers, and smart load control to increase autonomy [82]; 2) Adaptive thermal management uses phase change media, liquid loops, and passive radiative paths to widen operating envelopes [59]; 3) Mechanical hardening adds active damping, solid state storage, and isolation mounts to improve reliability; 4) Environmental protection with pressurized, hermetic, and pressure compen-

sated enclosures, plus anti corrosion treatments such as conformal coatings, specialized alloys, and electrochemical protection, extends service life; 5) High density packaging employs three dimensional integration, immersion cooling, and modular designs to maximize compute in constrained volumes [83, 84].

Platform Layer. Control plane innovations enable resilient operations across vehicular, underwater, space, industrial IoT, and microgrid systems: 1) Partition tolerant protocols sustain progress through consensus, replication, and eventual consistency under fragmentation; 2) Abstraction frameworks unify heterogeneous resources through containers, virtualization, and intelligent pooling; 3) Autonomous lifecycle management leverages satellite coordination, out of band control, and self healing; 4) Secure update mechanisms use dual partitions, verified boot, and incremental patches; 5) Isolation technologies combine hardware extensions, secure containers, and temporal partitions for predictable performance; 6) Cryptographic trust anchors with HSMs, TEEs, and remote attestation establish verifiable system integrity.

Application Layer. Application layer innovations address edge specific needs: 1) Inertial stabilization fuses multi sensor streams with Kalman filtering and predictive compensation to sustain accuracy under motion; 2) Intelligent edge processing reduces bandwidth through federated learning, local inference, and semantic compression; 3) Real time execution ensures temporal predictability with time triggered architectures, specialized operating systems, and hardware assisted scheduling [6]; 4) Safety assurance advances certification readiness through redundant execution, continuous monitoring, and formal verification; 5) Privacy preserving

computation upholds data sovereignty with homomorphic encryption, secure multi party protocols, and trusted execution; 6) efficient model adaptation balances accuracy and resource use through quantization, architecture search, and specialized frameworks [85].

Cross Layer. Integrated optimization strategies deliver system wide gains: 1) Coordinated energy management aligns voltage and frequency scaling, workload migration, and thermal control to co optimize power, thermal, and performance in vehicular, microgrid, and space systems; 2) Resilience architectures maintain availability in industrial IoT, underwater, and space through Byzantine fault tolerance, checkpoint recovery, and diversified redundancy; 3) Efficient observability preserves visibility under bandwidth limits in vehicular, edge, and underwater settings through sketch based telemetry, distributed tracing, and synthetic monitoring; 4) Adaptive resource orchestration maximizes scarce compute, storage, and network on vehicular, underwater, and space platforms using hierarchical control, game theoretic scheduling, and machine learning based optimization [5, 86].

7 Conclusion

μ DCs are rapidly transforming the landscape of distributed computing, offering unprecedented flexibility, efficiency, and autonomy in edge environments. Through a comprehensive survey spanning the infrastructure, platform, and application layers, this work has elucidated the critical design factors, technical trade-offs, and deployment paradigms that shape the current and future evolution of μ DC technologies.

At the infrastructure level, μ DCs are characterized by their modular architectures, heterogeneous compute units, adaptable networking, and

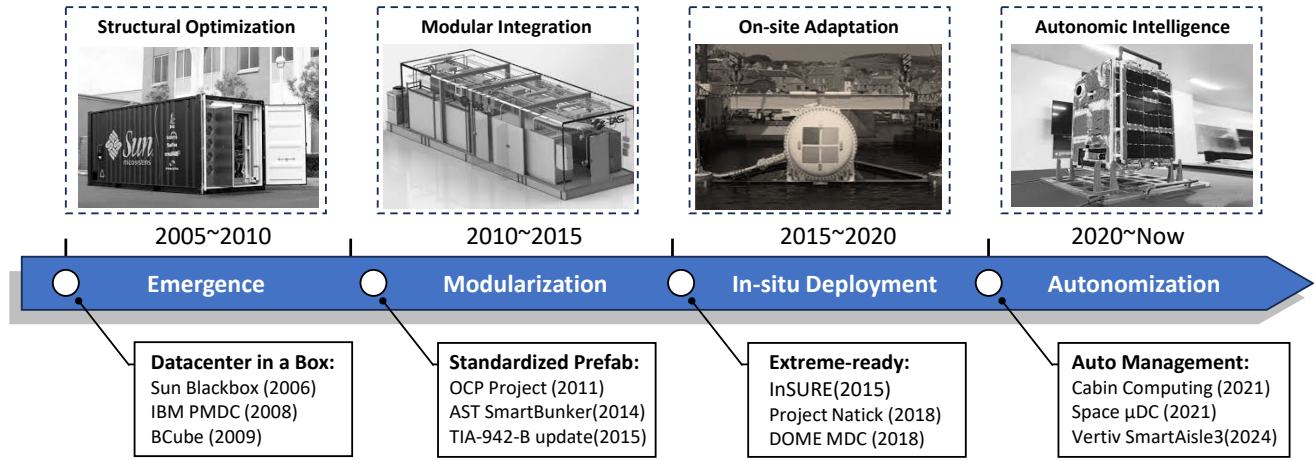


Fig. 3 The timeline of μ DCs evolution.

resilient energy and cooling solutions—each tailored to address the diverse and often extreme conditions of edge deployments. Platform innovations, from advanced resource management middleware to hybrid virtualization and multi-tenancy models, further enable μ DCs to maximize performance, scalability, and operational reliability in dynamic, distributed contexts. On the application front, μ DCs demonstrate strong capability to support both latency-critical and best-effort workloads, leveraging hardware–software co-design, cross-layer optimization, and scenario-specific scheduling techniques to meet stringent requirements for responsiveness, energy efficiency, and reliability.

Despite these advances, the field continues to face significant challenges, including ensuring infrastructure robustness under harsh conditions, reducing platform management complexity, and balancing competing application-level objectives. The integration of adaptive, intelligent orchestration across layers, alongside advances in federated management and distributed intelligence, will be pivotal for the next generation of μ DCs.

The evolution of μ DCs over the past two decades follows a clear progression from containerized pro-

totypes to autonomous, intelligent platforms. As shown in Figure 3, we categorize the development of μ DC into four typical phases. Early initiatives, such as Sun Blackbox, introduced the concept of a portable data center in a box and sparked practical interest. Standardization then gained momentum through the Open Compute Project and systems like AST SmartBunker, which set modular prefabrication practices and enabled scalable deployment. A subsequent phase established feasibility in extreme environments, with Microsoft’s underwater Project Natick and the ruggedized DOME μ DC systems demonstrating reliability under harsh thermal, mechanical, and maintenance constraints. Building on these foundations, current platforms represented by Cabin Computing [87] and Vertiv SmartAisle 3 integrate intelligent control, AI-driven management, and higher degrees of autonomous operation.

Looking ahead, development is likely to accelerate as adaptive infrastructure, intelligent middleware, and edge native applications continue to mature. The realization of large-scale autonomous μ DC networks that support cross-layer co-optimization and self-adaptive operation can

unlock new opportunities across industry, research, and society. Continued innovation and collaboration will be essential to close remaining gaps and to fully capture the transformative potential of μ DCs in the emerging edge computing era.

Acknowledgements We sincerely thank all the anonymous reviewers for their valuable comments that helped us to improve the paper. This work is supported in part by the National Key R&D Program of China (No. 2022YFB4501702) and the National Natural Science Foundation of China (No. U24A20234). The corresponding author is Chao Li.

References

1. Corporation N. Nvidia drive thor. <https://www.nvidia.com/en-us/self-driving-cars/drive-platform/drive-thor/>, 2024. Accessed: 2025-05-21
2. Liu L, Feng J, Mu X, Pei Q, Lan D, Xiao M. Asynchronous deep reinforcement learning for collaborative task computing and on-demand resource allocation in vehicular edge computing. *IEEE Transactions on Intelligent Transportation Systems*, 2023, 24(12): 15513–15526
3. Luo Q, Li C, Luan T H, Shi W. Minimizing the delay and cost of computation offloading for vehicular edge computing. *IEEE Transactions on Services Computing*, 2021, 15(5): 2897–2909
4. Shi J, Du J, Shen Y, Wang J, Yuan J, Han Z. Drl-based v2v computation offloading for blockchain-enabled vehicular networks. *IEEE Transactions on Mobile Computing*, 2022, 22(7): 3882–3897
5. Sun L, Hou X, Li C, Liu J, Wang X, Chen Q, Guo M. A2: Towards accelerator level parallelism for autonomous micromobility systems. *ACM Transactions on Architecture and Code Optimization*, 2024, 21(4): 1–20
6. Sun L, Li C, Hou X, Huang T, Xu C, Wang X, Bao G, Sun B, Rui S, Guo M. Jigsaw: Taming bev-centric perception on dual-soc for autonomous driving. In: 2024 IEEE Real-Time Systems Symposium (RTSS). 2024, 280–293
7. Talpes E, Williams D, Sarma D D. Dojo: The microarchitecture of tesla's exa-scale computer. In: 2022 IEEE Hot Chips 34 Symposium (HCS). 2022, 1–28
8. Saini L M, Ajmani P, Asha V, Pithamber K, Sreevani N. Kubeedge for scalable iot applications. In: 2024 7th International Conference on Contemporary Computing and Informatics (IC3I). 2024, 1–7
9. Macenski S, Foote T, Gerkey B, Lalancette C, Woodall W. Robot operating system 2: Design, architecture, and uses in the wild. *Science robotics*, 2022, 7(66): eabm6074
10. Chang W, Wu J. Fog/edge computing for security, privacy, and applications. volume 83. Springer, 2021
11. Dar F, Liyanage M, Radeta M, Yin Z, Zuniga A, Kosta S, Tarkoma S, Nurmi P, Flores H. Upscaling fog computing in oceans for underwater pervasive data science using low-cost micro-clouds. *ACM Transactions on Internet of Things*, 2023, 4(2): 1–29
12. Periola A, Alonge A, Ogudo K. Heat wave resilient systems architecture for underwater data centers. *Scientific Reports*, 2022, 12(1): 17161
13. Simon K. Project natick-microsoft's self-sufficient underwater datacenters. *IndraStra Global*, 2018, 4(6): 4
14. Elhoseny M, Lakhan A, Rashid A, Mohammed M, Abdulkareem K. Underwater sensor multi-parameter scheduling for heterogenous computing nodes. *ACM Transactions on Sensor Networks (TOSN)*, 2022, 18(3): 1–23
15. Lin C, Han G, Jiang J, Li C, Shah S B H, Liu Q. Underwater pollution tracking based on software-defined multi-tier edge computing in 6g-based underwater wireless networks. *IEEE Journal on Selected Areas in Communications*, 2023, 41(2): 491–503
16. Bleier N, Mubarik M H, Swenson G R, Kumar R. Space microdatacenters. In: *Proceedings of the 56th Annual IEEE/ACM International Symposium on Microarchitecture, MICRO 2023, Toronto, ON, Canada, 28 October 2023 - 1 November 2023*. 2023, 900–915
17. Bleier N, Eason R, Lembeck M, Kumar R. Architecting space microdatacenters: A system-level approach. In: *2025 IEEE International Symposium on High Performance Computer Architecture (HPCA)*. 2025, 1304–1319
18. Denby B, Lucia B. Orbital edge computing: Machine inference in space. *IEEE Computer Architecture Letters*, 2019, 18(1): 59–62
19. Denby B, Lucia B. Orbital edge computing: Nanosatellite constellations as a new class of computer system. In: *Proceedings of the Twenty-Fifth International Conference on Architectural Support for Programming Languages and Operating Systems*. 2020, 939–954
20. Wang S, Li Q, Xu M, Ma X, Zhou A, Sun Q. Tian-suan constellation: An open research platform. In: *2021 IEEE International Conference on Edge Computing (EDGE)*. 2021, 94–101
21. Yang C, Yuan J, Wu Y, Sun Q, Zhou A, Wang S, Xu M. Communication-efficient satellite-ground federated learning through progressive weight quantization.

- IEEE Transactions on Mobile Computing, 2024, 23(9): 8999–9011
22. Li Y, Li H, Liu W, Liu L, Chen Y, Wu J, Wu Q, Liu J, Lai Z. A case for stateless mobile core network functions in space. In: Proceedings of the ACM SIGCOMM 2022 Conference. 2022, 298–313
23. Azimi R, Fox T, Gonzalez W, Reda S. Scale-out vs scale-up: A study of arm-based socs on server-class workloads. ACM Trans. Model. Perform. Evaluation Comput. Syst., 2018, 3(4): 18:1–18:23
24. Xu D, Xu M, Lou C, Zhang L, Huang G, Jin X, Liu X. Socflow: Efficient and scalable DNN training on soc-clustered edge servers. In: Gupta R, Abu-Ghazaleh N B, Musuvathi M, Tsafirir D, eds, Proceedings of the 29th ACM International Conference on Architectural Support for Programming Languages and Operating Systems, Volume 1, ASPLOS 2024, La Jolla, CA, USA, 27 April 2024– 1 May 2024. 2024, 368–385
25. Ichnowski J, Chen K, Dharmarajan K, Adebola S, Danielczuk M, Mayoral-Vilches V, Zhan H, Xu D, Ghassemi R, Kubiawicz J, others . Fogros 2: An adaptive and extensible platform for cloud and fog robotics using ros 2. In: Proceedings of the IEEE International Conference on Robotics and Automation (ICRA). 2023
26. Fernandez M, others . Hpe spaceborne computer-2: Edge computing in space. In: Proceedings of the IEEE Aerospace Conference. 2022
27. NVIDIA Corporation . Nvidia drive agx: Autonomous vehicle computing platform, 2022
28. Wang L, others . Flash storage optimizations for edge systems. In: Proceedings of the 2021 USENIX Annual Technical Conference (ATC). 2021
29. Wang S, Zhao Y, Xu J, Yuan J, Hsu C H. Edge server placement in mobile edge computing. Journal of Parallel and Distributed Computing, 2019, 127: 160–168
30. Wu G, Chen X, Gao Z, Zhang H, Yu S, Shen S. Privacy-preserving offloading scheme in multi-access mobile edge computing based on madrl. Journal of Parallel and Distributed Computing, 2024, 183: 104775
31. Cao B, Fan S, Zhao J, Tian S, Zheng Z, Yan Y, Yang P. Large-scale many-objective deployment optimization of edge servers. IEEE Transactions on Intelligent Transportation Systems, 2021, 22(6): 3841–3849
32. Ning Z, Yang Y, Wang X, Guo L, Gao X, Guo S, Wang G. Dynamic computation offloading and server deployment for uav-enabled multi-access edge computing. IEEE Transactions on Mobile Computing, 2021, 22(5): 2628–2644
33. Ning Z, Yang Y, Wang X, Song Q, Guo L, Jamalipour A. Multi-agent deep reinforcement learning based uav trajectory optimization for differentiated services. IEEE Transactions on Mobile Computing, 2023, 23(5): 5818–5834
34. Xu X, Shen B, Yin X, Khosravi M R, Wu H, Qi L, Wan S. Edge server quantification and placement for offloading social media services in industrial cognitive iov. IEEE Transactions on Industrial Informatics, 2020, 17(4): 2910–2918
35. Tong Z, Deng X, Ye F, Basodi S, Xiao X, Pan Y. Adaptive computation offloading and resource allocation strategy in a mobile edge computing environment. Information Sciences, 2020, 537: 116–131
36. Bruschi R, Davoli F, Lombardo C, Sanchez O R. Evaluating the impact of micro-data center (μ dc) placement in an urban environment. In: 2018 IEEE Conference on Network Function Virtualization and Software Defined Networks (NFV-SDN). 2018, 1–7
37. Bruschi R, Davoli F, Lago P, Pajo J F. A multi-clustering approach to scale distributed tenant networks for mobile edge computing. IEEE Journal on Selected Areas in Communications, 2019, 37(3): 499–514
38. Bruschi R, Pajo J F, Davoli F, Lombardo C. Managing 5g network slicing and edge computing with the matilda telecom layer platform. Computer Networks, 2021, 194: 108090
39. Teng F, Ban Z, Li T, Sun Q, Li Y. A privacy-preserving distributed economic dispatch method for integrated port microgrid and computing power network. IEEE Transactions on Industrial Informatics, 2024
40. Switzer J, Marcano G, Kastner R, Pannuto P. Junkyard computing: Repurposing discarded smartphones to minimize carbon. In: Aamodt T M, Jerger N D E, Swift M M, eds, Proceedings of the 28th ACM International Conference on Architectural Support for Programming Languages and Operating Systems, Volume 2, ASPLOS 2023, Vancouver, BC, Canada, March 25–29, 2023. 2023, 400–412
41. Hu S, Qu Z, Tang B, Ye B, Li G, Shi W. Joint service request scheduling and container retention in serverless edge computing for vehicle-infrastructure collaboration. IEEE Transactions on Mobile Computing, 2023, 23(6): 6508–6521
42. Mobileye . Mobileye: Advanced driver-assistance and autonomous driving technologies. <https://www.mobileye.com/>, 2024. Accessed: 2025-05-21
43. Electric S. Micro data center r-series. <https://www.apc.com/us/en/solutions/business-solutions/micro-data-centers/r-series.jsp>, 2024. Accessed: 2025-05-21
44. Technologies D. Dell emc vxrail data sheet, 2021. Accessed: 2024-08

45. Hu Y, Lin X, Wang H, He Z, Yu X, Zhang J, Yang Q, Xu Z, Guan S, Fang J, Shang H, Tang X, Dai X, Wei S, Yin S. Wafer-scale computing: Advancements, challenges, and future perspectives [feature]. *IEEE Circuits and Systems Magazine*, 2024, 24(1): 52–81
46. Services A W. Aws wavelength: Mec infrastructure for 5g networks. <https://aws.amazon.com/wavelength/>, 2024. Accessed: 2025-05-21
47. Inc. V C. Verizon 5g edge. <https://www.verizon.com/business/products/5g-edge/>, 2024. Accessed: 2025-05-21
48. Inc. L. Liquidstar - solar-powered micro data centers. <https://www.liquidstar.io/>, 2023
49. Inc. S C. Subsea cloud: Underwater edge infrastructure. <https://www.subseacloudpartners.com/>, 2024. Accessed: 2025-05-21
50. Research M. Microsoft project natick: Underwater datacenter initiative. <https://natick.research.microsoft.com/>, 2024. Accessed: 2025-05-21
51. Foundry E. Edgex foundry: The open source edge platform. <https://www.edgexfoundry.org/>, 2024. Accessed: 2025-05-21
52. Acun B, Lee B, Kazhamiaka F, Maeng K, Gupta U, Chakkaravarthy M, Brooks D, Wu C J. Carbon explorer: A holistic framework for designing carbon aware datacenters. In: *Proceedings of the 28th ACM International Conference on Architectural Support for Programming Languages and Operating Systems*, Volume 2. 2023, 118–132
53. Li C, Hu Y, Zhou R, Liu M, Liu L, Yuan J, Li T. Enabling datacenter servers to scale out economically and sustainably. In: *Proceedings of the 46th annual IEEE/ACM international symposium on microarchitecture*. 2013, 322–333
54. Kontorinis V, Zhang L E, Aksanli B, Sampson J, Homayoun H, Pettis E, Tullsen D M, Rosing T S. Managing distributed ups energy for effective power capping in data centers. *ACM SIGARCH Computer Architecture News*, 2012, 40(3): 488–499
55. Pal S, Liu J, Alam I, Cebry N, Suhail H, Bu S, Iyer S S, Pamarti S, Kumar R, Gupta P. Designing a 2048-chiplet, 14336-core waferscale processor. In: *2021 58th ACM/IEEE Design Automation Conference (DAC)*. 2021, 1183–1188
56. Lie S. Cerebras architecture deep dive: First look inside the hw/sw co-design for deep learning: Cerebras systems. In: *2022 IEEE Hot Chips 34 Symposium (HCS)*. 2022, 1–34
57. Min D, Byun I, Lee G h, Kim J. Cooldc: A cost-effective immersion-cooled datacenter with workload-aware temperature scaling. *ACM Transactions on Architecture and Code Optimization*, 2024, 21(3): 1–27
58. Qouneh A, Li C, Li T. A quantitative analysis of cooling power in container-based data centers. In: *2011 IEEE International Symposium on Workload Characterization (IISWC)*. 2011, 61–71
59. Pei Q, Chen S, Zhang Q, Zhu X, Liu F, Jia Z, Wang Y, Yuan Y. Cooledge: hotspot-relievable warm water cooling for energy-efficient edge datacenters. In: *Proceedings of the 27th ACM International Conference on Architectural Support for Programming Languages and Operating Systems*. 2022, 814–829
60. Li H, Bai Z, Li Z, Mitra T. Asadi: Accelerating sparse attention using diagonal-based in-situ computing. In: *2024 IEEE International Symposium on High-Performance Computer Architecture (HPCA)*. 2024
61. Toosi A N, Son J, Buyya R. Clouds-pi: A low-cost raspberry-pi based micro data center for software-defined cloud computing. *IEEE Cloud Computing*, 2018, 5(5): 81–91
62. Veen A, others . Dome: Micro data center architecture for the ska radio telescope. In: *IEEE Cluster*. 2018
63. Yuan G, Behnam P, Li Z, Shafiee A, Lin S, Ma X, Liu H, Qian X, Bojnordi M N, Wang Y, others . Forms: Fine-grained polarized reram-based in-situ computation for mixed-signal dnn accelerator. In: *2021 ACM/IEEE 48th Annual International Symposium on Computer Architecture (ISCA)*. 2021, 265–278
64. Song M, Zhong K, Zhang J, Hu Y, Liu D, Zhang W, Wang J, Li T. In-situ ai: Towards autonomous and incremental deep learning for iot systems. In: *2018 IEEE International Symposium on High Performance Computer Architecture (HPCA)*. 2018, 92–103
65. Zhou J, Cao K, Zhou X, Chen M, Wei T, Hu S. Throughput-conscious energy allocation and reliability-aware task assignment for renewable powered in-situ server systems. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, 2021, 41(3): 516–529
66. Li C, Hu Y, Liu L, Gu J, Song M, Liang X, Yuan J, Li T. Towards sustainable in-situ server systems in the big data era. *Acm Sigarch Computer Architecture News*, 2015, 43(3S): 14–26
67. Foundation O. Starlingx: Open source edge cloud platform. <https://www.starlingx.io/>, 2022
68. Abuibaid M, Ghorab A H, Seguin-Mcpeake A, Yuen O, Yungblut T, St-Hilaire M. Edge workloads monitoring and failover: a starlingx-based testbed implementation and measurement study. *IEEE Access*, 2022, 10: 97101–97116
69. Li C, Xue Y, Wang J, Zhang W, Li T. Edge-oriented computing paradigms: A survey on architecture design

- and system management. *ACM Computing Surveys (CSUR)*, 2018, 51(2): 1–34
70. Foundation O N. Aether: Open source 5g connected edge cloud platform. <https://www.opennetworking.org/aether/>, 2021
 71. Services A W. Aws snowball edge. <https://aws.amazon.com/snowball-edge/>, 2022
 72. NVIDIA . Nvidia drive agx platform. <https://www.nvidia.com/en-us/self-driving-cars/drive-platform/>, 2022
 73. Labs R. K3s for smart surveillance. <https://rancher.com/docs/k3s/latest/en/>, 2022
 74. Intel . Intel openness: Open network edge services software. <https://www.openness.org/>, 2022
 75. Project E. Edgefabric: Decentralized distributed sgd framework. <https://github.com/edgefabric/>, 2022
 76. Zhang L, Feng W, Li C, Hou X, Wang P, Wang J, Guo M. Tapping into nfv environment for opportunistic serverless edge function deployment. *IEEE Transactions on Computers*, 2021, 71(10): 2698–2704
 77. Li Z, Lou J, Wu J, Guo J, Tang Z, Shen P, Jia W, Zhao W. Online container scheduling with fast function startup and low memory cost in edge computing. *IEEE Transactions on Computers*, 2024
 78. Xiao K, Yang S, Li F, Zhu L, Chen X, Fu X. Making serverless not so cold in edge clouds: A cost-effective online approach. *IEEE Transactions on Mobile Computing*, 2024, 23(9): 8789–8802
 79. Liu D, Zhang L, Xu Y, Wang X, Sun L, Pu Y, Hou X, Li C, Guo M. Power synchronization: taming massive diversified serverless functions under power constraints. *Science China Information Sciences*, 2025, 68(3): 132101
 80. Xu Z, Zhou L, Liang W, Xia Q, Xu W, Ren W, Ren H, Zhou P. Stateful serverless application placement in mec with function and state dependencies. *IEEE Transactions on Computers*, 2023, 72(9): 2701–2716
 81. Li Y, Zeng D, Gu L, Ou M, Chen Q. On efficient zygot container planning toward fast function startup in serverless edge cloud. In: *IEEE INFOCOM 2023-IEEE Conference on Computer Communications*. 2023, 1–9
 82. Hou X, Xu T, Li C, Xu C, Liu J, Hu Y, Zhao J, Leng J, Cheng K T, Guo M. A tale of two domains: Exploring efficient architecture design for truly autonomous things. In: *2024 ACM/IEEE 51st Annual International Symposium on Computer Architecture (ISCA)*. 2024, 167–181
 83. Hong H, Wu Q, Dong F, Song W, Sun R, Han T, Zhou C, Yang H. Netgraph: An intelligent operated digital twin platform for data center networks. In: *Proceedings of the ACM SIGCOMM 2021 workshop on network-application integration*. 2021, 26–32
 84. Sarkar S, Naug A, Guillen A, Luna R, Gundecha V, Babu A R, Mousavi S. Sustainability of data center digital twins with reinforcement learning. In: *Proceedings of the AAAI Conference on Artificial Intelligence*. 2024, 23832–23834
 85. Chi Y, Yue J, Liao X, Liu H, Jin H. A hybrid memory architecture supporting fine-grained data migration. *Frontiers of Computer Science*, 2024, 18(2): 182103
 86. Tian C, Liu H, Liao X, Jin H. Ucat: heterogeneous memory management for unikernels. *Frontiers of Computer Science*, 2023, 17(1): 171204
 87. Jiang C, Ding Z, Yu J, others . Cabin computing. *Sci Sin Inform*, 2021, 51: 1233–1254. in Chinese



Jinyang Guo received a master's degree in computer science and technology from Shanghai Jiao Tong University. He is working toward a PhD. degree in the School of Computer Science, Shanghai Jiao Tong University.

His main research interests include MDC, distributed collaboration, and edge computing.



Chao Li received his Ph.D. degree from the University of Florida in 2014. He is currently a professor with tenure in the School of Computer Science, Shanghai Jiao Tong University. His research mainly focuses on computer architecture

and systems for emerging applications. He is a Distinguished Member of CCF.



Mingxuan Zhang received his bachelor's degree in Software Engineering from Southeast University, China. He is currently pursuing a master's degree in the School of Computer Science at Shanghai Jiao Tong University, China. His

main research interests focus on emerging LLM inference and serverless computing, especially on integrating the two in a unified computing framework.



Minyi Guo received his Ph.D. degree in computer science from the University of Tsukuba, Japan. He is a Zhiyuan Chair Professor at Shanghai Jiao Tong University.

His research interests include parallel and distributed computing, compiler optimizations, and cloud computing. He is an IEEE Fellow and a CCF Fellow.



Yunwei Li received the BS degree from Shanghai Jiao Tong University, China. He is working toward a master degree in Shanghai Jiao Tong University Paris Elite Institute of Technology(SPEIT),

China. His main research interests include ML reliability and edge computing.