AVNET®

WHY NETWORK SPEED & INTELLIGENCE ARE MUST-HAVES FOR AI-ENABLED INFRASTRUCTURE

FEBRUARY 2026

TABLE OF CONTENTS

EXECUTIVE SUMMARY

How intelligent networks define realized AI performance

AI workloads have redrawn the limits of infrastructure performance

Training and inference no longer stress compute alone—they reveal the limits of how data moves, synchronizes, and scales across connected systems. As GPU clusters grow more dense and workloads more distributed, the network has become both the bottleneck and the opportunity. In the AI era, throughput, latency, and flow control determine whether expensive accelerators operate at full capacity or sit idle waiting for data.

From utility to differentiator

For decades, networks were treated as plumbing—reliable, standardized, and largely invisible once provisioned. That model no longer holds. The modern AI data center depends on continuous, low-latency data movement between thousands of devices. The network is now a performance component in its own right, influencing model-training time, inference speed, and operational efficiency.

Flows, not nodes, define the new performance frontier

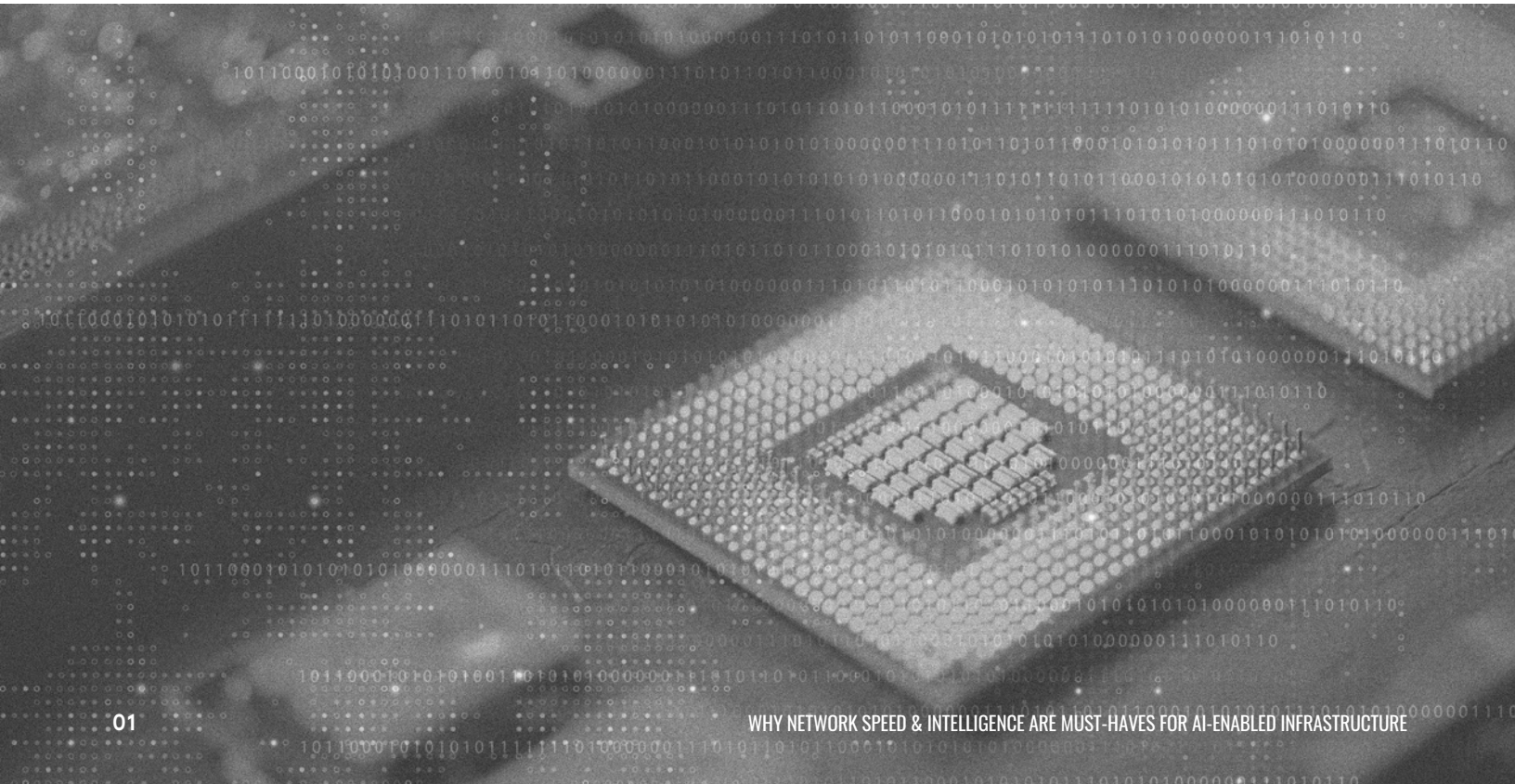
Optimizing AI data centers by focusing on flows instead of nodes represents a shift from managing individual network devices to dynamically managing the actual data traffic that flows between them. Unlike traditional workloads, AI generates massive bursts of east-west traffic (server-to-server) within the data center, making congestion a critical performance constraint. Engineers who once tuned nodes must now tune the pathways that connect them.

Intelligence amplifies speed

Raw bandwidth alone can't solve these challenges. Intelligent fabrics that are built with real-time telemetry, adaptive routing, and automation, convert link speed into usable performance by steering traffic where it's needed most. In this way, the network evolves from static connectivity to an active, self-optimizing system.

What this paper explores

This white paper examines how network speed and intelligence together determine AI infrastructure efficiency. It analyzes the technical forces reshaping data-center design, highlights real-world approaches such as GPU fabrics, optical interconnects, and adaptive routing, and outlines practical steps toward self-optimizing networks. The goal is straightforward: to help engineers build systems where the network no longer limits compute potential but accelerates it.



THE SHIFTING ROLE OF NETWORKING IN AI INFRASTRUCTURE

Why the network has become the new performance frontier

From compute-centric to data-centric performance

AI workloads are changing what "fast" means inside the data center. In traditional environments, performance gains came from scaling CPU or GPU power—optimizing the node. In AI environments, especially distributed training, the limiting factor is no longer inside the server. It's between them. The performance boundary has moved into the network itself.

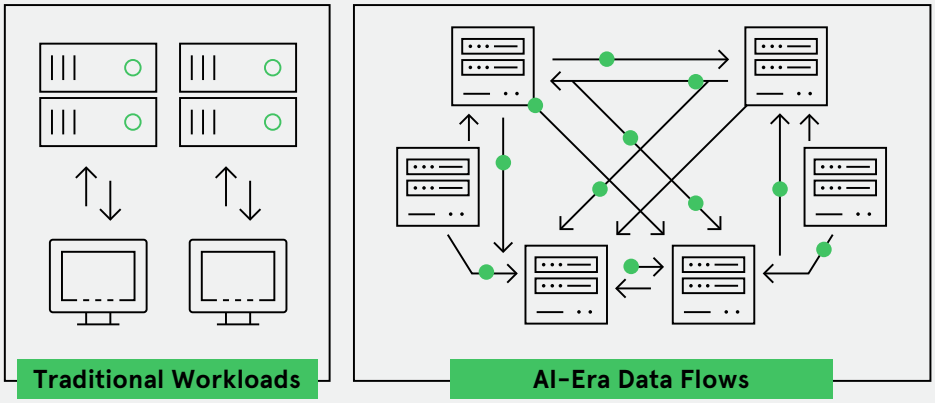
The flow-based paradigm

Optimizing AI data centers by focusing on flows instead of nodes represents a shift from managing individual network devices to dynamically managing the actual data traffic that flows between them. As previously mentioned, the massive bursts of east-west traffic makes congestion a serious threat to performance.

This shift requires engineers to think in terms of data movement, not device management. Traffic must be observed, prioritized, and intelligently steered in real time to keep GPU clusters saturated and synchronized. In effect, performance optimization has moved from the device layer to the data plane.

"Flows, Not Nodes" — How AI Traffic Redefines the Network

AI training floods the fabric with east-west data movement—making flow control the new frontier of performance.



East–west traffic: the workload multiplier

AI training floods the network with short-lived, high-intensity data exchanges—gradients, activations, and model updates flowing continuously between GPUs. Inference creates continuous data movement across compute, caching, and storage systems—especially in retrieval-augmented generation (RAG) architectures.

This east-west pattern amplifies any inefficiency. Even the smallest pockets of congestion or latency ripple across nodes, compounding delays and reducing GPU utilization. In practice, overall system performance is limited not by compute capacity but by how efficiently data moves through the network.

Visibility and automation have become prerequisites

Static configurations can't adapt quickly enough to these dynamic patterns. To maintain consistent throughput, engineers need real-time telemetry—visibility into microbursts, queue depth, and packet latency. That visibility feeds the next layer: automation. Policy-driven controls and orchestration frameworks (e.g., Ansible, Terraform, or comparable intent-based controllers) allow networks to adjust routing and QoS dynamically, enforcing intent rather than device-level settings.

Why it matters now

AI's growth curve has forced a mindset change. Where traditional data centers valued predictability, AI-driven infrastructures demand adaptability. No longer a background service, the network is a living component that determines the system's total efficiency. Get the network right, and you scale faster. Get it wrong, and GPUs sit idle.

/ WHAT'S ON A NETWORK & COMMUNICATIONS ENGINEER'S MIND (AI EDITION)

Seven realities shaping AI-era infrastructure decisions

LATENCY AND EAST–WEST TRAFFIC DOMINANCE	Industry estimates suggest that by 2025, AI workloads could represent up to one-third of data center traffic, largely driven by east-west flows between GPU clusters. Supporting survey data indicate that over half of data center operators expect AI to overtake traditional cloud workloads in the next 2-3 years.
SCALABILITY UNDER BURST CONDITIONS	AI training creates unpredictable surges in data transfer—high-volume bursts when models synchronize or checkpoints are saved. Network architectures have to absorb these transient loads without packet loss or queue buildup. Designing for sustained peak efficiency rather than average throughput has become table stakes.
POWER AND THERMAL EFFICIENCY	As high-bandwidth fabrics and optical links multiply, power consumption rises sharply. Every additional watt consumed by the network competes with GPU cooling budgets and facility power envelopes. Engineers are being asked to balance bandwidth growth with strict power efficiency targets.
OBSERVABILITY AND AUTOMATION	The scale of AI fabrics makes manual troubleshooting impossible. Engineers depend on real-time telemetry—flow analytics, microburst detection, and packet latency insights—to understand dynamic conditions. Automation frameworks convert that visibility into action, applying policies and remediating issues automatically.
SECURITY AND DATA ISOLATION	AI systems often process proprietary or regulated data across shared infrastructure. Engineers must isolate workloads while maintaining low-latency communication. Network segmentation, encryption in transit, and policy-based microsegmentation have become as essential as throughput optimization.
HARDWARE SUPPLY AND LEAD-TIME RISK	Building or upgrading large-scale AI clusters requires specialized switches, optics, and cables that are subject to long lead times. Engineers must design with parts availability in mind—identifying multi-vendor options or modular architectures to avoid future bottlenecks.
CLOUD AND EDGE INTEGRATION	Hybrid AI workflows increasingly span on-prem data centers, public clouds, and edge locations. Consistent connectivity and orchestration across these domains are critical. Engineers are tasked with designing fabrics that extend policy, observability, and performance guarantees across heterogeneous environments.

KEY INSIGHT / The role of the network engineer has shifted from provisioning connectivity to orchestrating performance. AI doesn't just need links—it needs intelligence embedded throughout the fabric.

WHY SPEED ALONE ISN'T ENOUGH – THE RISE OF INTELLIGENT NETWORKING

Turning bandwidth into usable performance

Bandwidth without intelligence leaves performance on the table

Adding faster links isn't the same as removing bottlenecks. While AI workloads need bandwidth, they also need coordination. When multiple GPUs train together, each step depends on thousands of simultaneous data exchanges. Without intelligence in the fabric—telemetry, routing awareness, congestion control—higher speeds simply move congestion from one link to another. In AI systems, the network's job isn't just to connect devices but to maintain synchronization between them.

GPU fabrics — synchronizing compute at memory speed

Modern GPU fabrics (NVIDIA NVLink/NVSwitch and AMD Infinity Fabric as examples) illustrate how networking has moved inside the server chassis. These interconnects now deliver terabit-scale bidirectional bandwidth per GPU, allowing multiple accelerators to operate as one logical compute system. This level of intra-node communication ensures that gradient updates and parameter exchanges happen as fast as the GPUs can compute them. In practice, it's the same principle that engineers now apply at rack or cluster scale: treat data movement as part of compute design, not an afterthought.

Example: The NVIDIA DGX H100 architecture delivers roughly 2.3× higher all-reduce performance than its predecessor—achieved through fabric redesign, not just faster GPUs.

High-bandwidth optical interconnects — scaling the fabric across racks

Inside AI clusters, optical interconnects form the spine that ties hundreds or thousands of GPUs together. 400 to 800 Gb/s optical modules—built on coherent DSPs and advanced laser technologies—enable low-latency, lossless Ethernet fabrics that preserve signal integrity over longer distances. Optical design choices—connector quality, wavelength multiplexing, and dispersion control—directly influence convergence time for distributed training jobs.

According to Cisco's Nexus 9000 Series for AI Clusters white paper (2024), "high-performance and low-latency Ethernet fabrics are essential for inter-GPU networks." Even with 800 Gb/s links, without proper path planning or adaptive congestion control, east-west traffic can still stall GPU pipelines.

Adaptive routing — intelligence in motion

Speed alone can't predict or prevent congestion. Modern AI fabrics rely on adaptive routing—using real-time telemetry to decide where traffic should go. Technologies such as NVIDIA Quantum-2 InfiniBand, Cisco AI Fabric, and Arista EOS AI Network Telemetry dynamically reroute packets away from busy paths. The network, in effect, becomes self-aware: it senses imbalance and corrects it before packet loss occurs.

This intelligence layer also enables intent-based performance tuning. Engineers can specify policies—prioritize gradient synchronization, preserve latency budgets for inference, or isolate test jobs—and the fabric enforces those objectives automatically. Adaptive routing transforms the network from a static transport medium into an active performance participant.

Bringing it together — the network as co-processor

GPU fabrics accelerate communication inside the node; optical interconnects extend that performance across racks; adaptive routing keeps it efficient in real time. Together they show that modern AI networks don't just carry data—they compute with it. The result is a fabric that acts like a distributed co-processor, translating raw bandwidth into consistent throughput, faster convergence, and measurable ROI.

THE BUSINESS CASE – ROI OF INTELLIGENT NETWORKS

Where speed and intelligence convert into measurable efficiency

The cost of idle compute

In AI infrastructure, underutilized GPUs are the most expensive inefficiency in the system. **A single eight-GPU server can consume more than 10 kW of power even when waiting for data.** Across hundreds of nodes, that energy burn adds up to millions in operating cost and delays model training cycles. Intelligent networks pay for themselves by keeping compute resources continuously fed—turning what would be idle time into productive processing.

Latency directly impacts training efficiency

Even small reductions in delay between GPUs translate into faster model convergence and higher hardware utilization. When network latency drops, synchronization steps complete sooner, allowing training jobs to progress at full compute speed. This effect compounds across thousands of iterations: **the shorter each communication cycle, the sooner the model reaches accuracy targets and the less energy it consumes per epoch.**

Case vignette — "A 10× link only helps if the data knows where to go."

High-speed optics alone don't guarantee performance. **Cisco's own AI-ready data-center deployment showed that increasing link speed without architectural redesign could not eliminate congestion**[1]. True scalability was achieved only after the team introduced a dedicated, lossless back-end network optimized for GPU-to-GPU traffic. The takeaway is clear: bandwidth improvements amplify existing design flaws unless they're paired with intelligent routing, segmentation, and congestion management.

Fast links move data; smart fabrics move it efficiently.

Energy and time savings through utilization

Higher GPU utilization shortens total training time and reduces per-job power draw. **Studies of distributed AI clusters show that a 5–10 percent gain in utilization can cut overall energy consumption by a similar margin, lowering both operational cost and carbon footprint**[2]. The same principles apply at inference scale, where low-latency interconnects reduce response time and allow consolidation of workloads onto fewer, more efficient nodes.

Network intelligence extends hardware life

Adaptive telemetry and policy-driven controls also reduce mechanical wear. **Smarter congestion management prevents overheating, minimizes switch fan speeds, and avoids excessive retransmissions**—all of which extend component lifespan. For operators facing 18-month lead times on high-bandwidth optics, that reliability translates directly into business continuity.

ROI in three dimensions

1. **Performance ROI**
   More throughput per GPU hour; shorter training cycles.
2. **Operational ROI**
   Lower energy use, fewer manual interventions, and reduced downtime.
3. **Asset ROI**
   Longer hardware service life and better utilization of capital-intensive infrastructure.

Intelligent networks do far more than just connect accelerators—they ensure those accelerators earn their keep. In the AI era, network performance and financial performance have effectively merged into the same metric: utilization.

[1] Cisco IT Case Study — "Building an AI-Ready Infrastructure" (Cisco on Cisco, 2024).

[2] NVIDIA, "Optimizing AI Infrastructure for Energy Efficiency," 2024; Uptime Institute, "AI and Data Center Energy Demand," 2024.

DESIGN PRINCIPLES FOR AI-READY NETWORKING

Building networks that adapt as fast as AI workloads evolve

1

Design for deterministic latency, not just bandwidth

Throughput matters, but predictability matters more. AI training and inference workloads depend on consistent, low-latency data movement between GPUs. Even with 400 or 800 Gb/s links, fluctuating delay can desynchronize training steps. Deterministic latency—achieved through congestion-free fabrics, precise queue management, and buffer-to-buffer flow control—keeps GPUs synchronized and utilization high.

2

Implement real-time telemetry and closed-loop visibility

You can't optimize what you can't observe. Modern AI fabrics should collect live data on packet delay, loss, and congestion at microsecond granularity. Streaming telemetry protocols (e.g., gNMI, sFlow, NetFlow v10) feed analytics engines that spot microbursts and reroute traffic before performance drops. Visibility must extend across physical, virtual, and optical layers to capture the full end-to-end path of AI traffic.

3

Automate through policy-driven control

Manual configuration can't keep pace with AI's dynamism. Policy-based automation—implemented through tools such as Ansible, Terraform, or equivalent network orchestration platforms—lets engineers define outcomes (e.g., latency budgets, QoS tiers, workload isolation) rather than individual switch commands. The system continuously enforces those intents across devices, reducing error and allowing the network to respond in real time to workload shifts.

4

Plan for hybrid and multi-cloud flexibility

AI workflows increasingly span private data centers, public cloud instances, and edge locations. Network design should assume mobility: data and compute will move between environments. Use consistent policy frameworks, encrypted overlays, and identity-based routing to maintain performance and security wherever the workload resides.

5

Engineer for sustainability and lifecycle efficiency

AI networking doesn't just drive performance—it drives power draw. Optics, cooling, and switching fabrics consume a significant share of total facility energy. Design choices that improve utilization and airflow efficiency directly lower operational cost and environmental impact. Continuous monitoring of power per Gbps or per training epoch ensures

Network Topology Maturity Model

From fixed connections to adaptive fabrics—the evolution of network intelligence.

STATIC	AUTOMATED	INTELLIGENT
Manually configured, limited visibility; suitable for predictable client-server workloads.	API-driven, repeatable playbooks; faster and less error-prone but still reactive.	Telemetry-fed, self-optimizing; adaptive routing and intent enforcement at scale.

The guiding principle

AI-ready networks must behave less like infrastructure and more like systems. Determinism, observability, automation, flexibility, and sustainability together form the design foundation for next-generation fabrics—networks capable of sensing, deciding, and adapting as fast as the workloads they support.

IMPLEMENTATION PATH – FROM STATIC TO SELF-OPTIMIZING

How to evolve existing networks for AI-scale performance

STEP 1

Instrumentation — Measure what matters

Modernization starts with visibility. Static networks lack the telemetry to diagnose congestion or latency drift, so the first step is embedding sensors and data collection at every layer. Streaming telemetry (gNMI, NetFlow v10, or sFlow) exposes microburst behavior, packet latency, and utilization trends in real time. Focus measurement on metrics that correlate directly to GPU efficiency: queue depth, buffer occupancy, and inter-GPU delay. Without these baselines, optimization efforts are guesswork.

STEP 2

Automation — Replace manual with repeatable

Once telemetry is in place, the next step is automation. Engineers move from CLI-based management to playbooks and templates using infrastructure-as-code tools such as Ansible, Terraform, or comparable orchestration systems. The goal is consistency and speed—deploying configurations in minutes rather than hours and eliminating human error. Policy-driven automation also allows iterative testing: rules can be adjusted, replayed, and verified against telemetry data. At this stage, the network is faster to configure but still follows predefined logic rather than adaptive behavior.

STEP 3

Intelligence — Close the loop

Intelligence begins when the network uses telemetry to adjust itself. Congestion-aware routing, queue balancing, and intent-based enforcement convert static automation into adaptive control. Technologies such as Cisco Nexus 9000 AI Fabric, Arista EOS AI Telemetry, or NVIDIA Quantum-2 InfiniBand enable feedback loops that sense and respond within milliseconds. Engineers define the policy ("maintain latency < 10 µs for synchronization traffic"), and the network automatically tunes paths and QoS to meet it. This is the point where the fabric transitions from managed infrastructure to self-optimizing system.

STEP 4

Validation — Quantify the gains

Every modernization step should produce measurable results. Track GPU utilization, average latency, and energy per training run before and after each phase. Small increments—5 percent higher utilization or 2 percent lower retransmission rate—compound quickly across large clusters. Validation data also strengthens the business case for continued investment, converting engineering metrics into ROI evidence.

STEP 5

Iteration — Design for continuous adaptation

AI workloads evolve faster than infrastructure refresh cycles. Treat network modernization as an ongoing process, not a one-time project. Use analytics feedback to refine automation policies, update telemetry models, and re-evaluate topology as cluster sizes grow. The most successful teams operate their network like software—versioned, tested, and continuously improved.

From playbooks to prediction

This path doesn't require a complete rebuild. By layering instrumentation, automation, and intelligence, existing networks can evolve into AI-ready fabrics incrementally. The payoff is measurable: fewer idle GPUs, faster model convergence, lower power draw, and a network that anticipates rather than reacts. The journey from static to self-optimizing begins with visibility—and ends with a system that learns from its own performance.

CONCLUSION

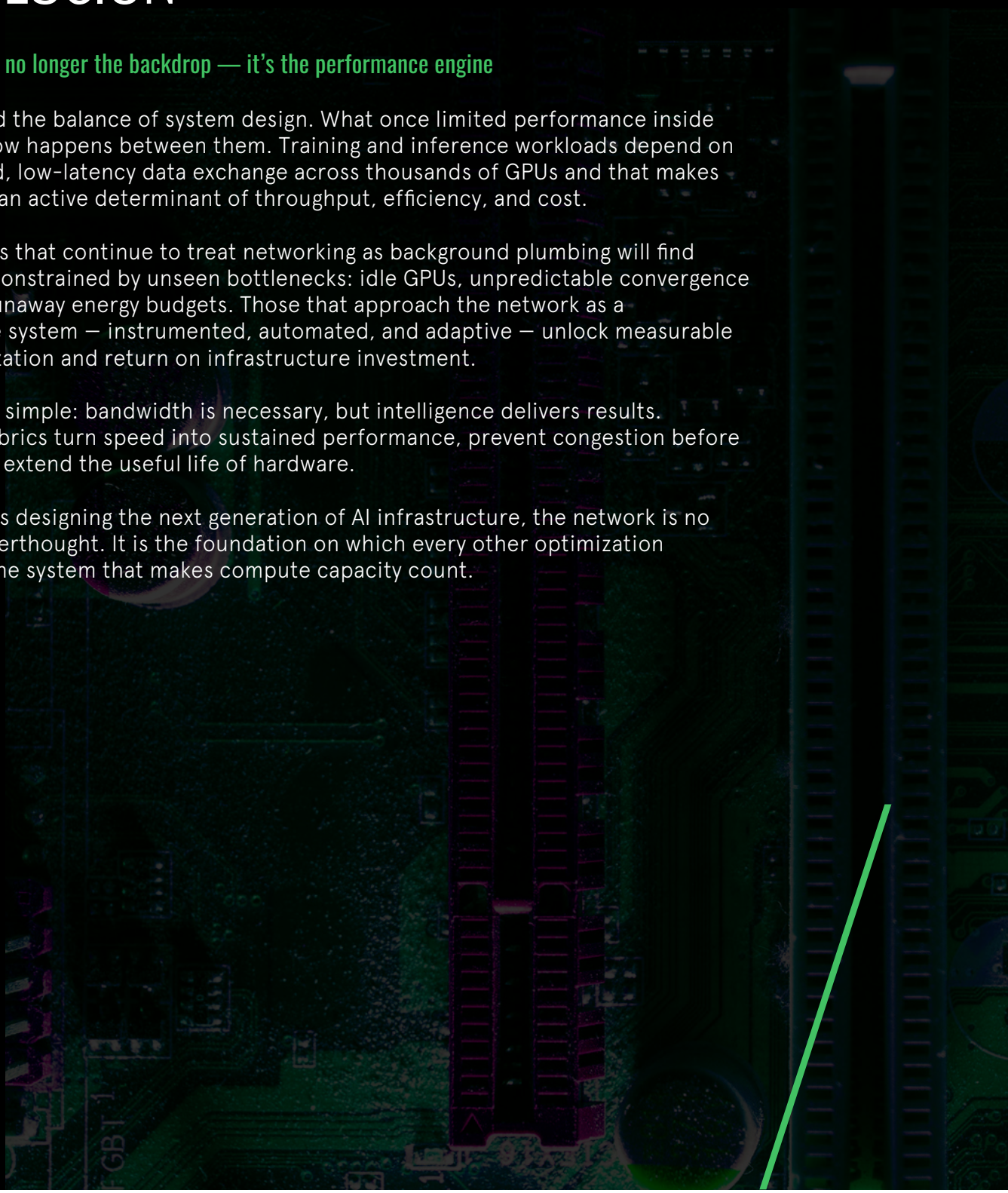
The network is no longer the backdrop — it's the performance engine

AI has shifted the balance of system design. What once limited performance inside the server now happens between them. Training and inference workloads depend on synchronized, low-latency data exchange across thousands of GPUs and that makes the network an active determinant of throughput, efficiency, and cost.

Organizations that continue to treat networking as background plumbing will find themselves constrained by unseen bottlenecks: idle GPUs, unpredictable convergence times, and runaway energy budgets. Those that approach the network as a performance system — instrumented, automated, and adaptive — unlock measurable gains in utilization and return on infrastructure investment.

The lesson is simple: bandwidth is necessary, but intelligence delivers results. Intelligent fabrics turn speed into sustained performance, prevent congestion before it starts, and extend the useful life of hardware.

For engineers designing the next generation of AI infrastructure, the network is no longer an afterthought. It is the foundation on which every other optimization depends — the system that makes compute capacity count.



GLOSSARY OF KEY TERMS

Adaptive Routing — A network routing method that uses real-time telemetry to detect congestion and dynamically select the least-loaded path, improving throughput and reducing latency during AI training and inference.

Automation (Policy-Driven) — The use of orchestration tools (e.g., Ansible, Terraform, Netris) to apply configurations and enforce policies automatically across devices, replacing manual CLI commands with repeatable, intent-based workflows.

Bandwidth vs. Latency — Bandwidth measures how much data can move per second; latency measures how long each transfer takes. AI performance depends more on low and consistent latency than on maximum bandwidth alone.

Deterministic Latency — The property of maintaining consistent delay under load; essential for synchronized GPU communication in distributed AI training.

East–West Traffic — Data that moves laterally between servers or GPUs within a data center. Dominant in AI workloads, contrasting with north–south traffic between clients and servers.

Inference Load — The real-time execution of trained AI models, where results must be generated quickly and efficiently—often across distributed edge or cloud environments.

Intent-Based Networking — A control model in which engineers define desired outcomes (e.g., latency targets, QoS levels), and the network automatically implements and maintains those intents through continuous monitoring and policy enforcement.

Lossless Ethernet Fabric — A network architecture using flow-control mechanisms (e.g., DCB, RoCEv2) to prevent packet loss during high-volume GPU data exchange, improving training stability and convergence speed.

Microburst — A sudden, millisecond-scale spike in network traffic that can overflow buffers, increase latency, and cause GPU idle time if undetected.

Network Telemetry — Continuous collection of real-time performance data such as delay, queue depth, or packet drops—used to diagnose congestion and inform automated responses.

North–South Traffic — Data movement between client devices (north) and data-center servers (south). Traditional enterprise workloads are dominated by this pattern.

Retrieval-Augmented Generation (RAG) — A technique that combines large language models with external data retrieval systems, generating continuous bidirectional traffic between compute, cache, and storage layers during inference.

Static vs. Dynamic Topology — A static network requires manual configuration and fixed routes; a dynamic or self-optimizing topology adjusts automatically based on telemetry, policy, or workload changes.

Utilization (Compute Utilization) — The percentage of time hardware (e.g., GPUs) actively processes workloads. Higher utilization directly improves ROI and reduces energy per training run.

ABOUT AVNET

As a leading global technology distributor and solutions provider, Avnet has served customers' evolving needs for more than a century. Through regional and specialized businesses around the world, we support customers and suppliers at every stage of the product lifecycle. We help companies adapt to change and accelerate the design and supply stages of product development. With a unique viewpoint from the center of the technology value chain, Avnet is a trusted partner that solves complex design and supply chain issues so customers can realize revenue faster.

Learn more about Avnet at www.avnet.com