

OPEN STANDARDS AND THE COST OF A LOCKED-IN NETWORK

Every layer of an AI cluster can be changed independently. Accelerators get chosen fresh at each purchase, and the platform you buy next year need not match the one you bought last year. Storage is procured on its own terms. Software gets versioned against interfaces, not hardware.

The network fabric wired through those racks breaks the pattern. It is chosen once per build, and it decides what the rest of the cluster is permitted to become. Change the accelerator and the fabric constrains the choice. Expand the cluster and the fabric sets the terms. It is the one decision that reaches into every other decision, and it draws the least scrutiny of any of them.

That gap matters more now than it did two years ago. Three shifts have reshaped AI data centers. Disaggregated inference became the primary workload. Clusters crossed the scale where congestion stops being an annoyance and hardens into a structural constraint. And the cost of wasted capital, once absorbed without much complaint, grew too large to overlook.

For years, building AI compute meant buying one vendor's GPUs. That is no longer true. The major hyperscalers now run their own custom silicon alongside merchant GPUs: Google is several generations into its TPU line. AWS builds both Trainium and Inferentia for training and inference. Microsoft has announced its Maia accelerator for inference, and Meta runs its MTIA silicon for ranking and inference.¹ The accelerator became a design choice, and the shift runs further than the market narrative admits. No single vendor will own the GPU pod, and the largest buyers have started spending real engineering money to keep it that way.

THE ACCELERATOR LANDSCAPE**CORNELIS NETWORKS**

No single vendor owns the GPU pod

Every major hyperscaler now runs custom silicon alongside merchant GPUs

Google CUSTOM SILICON TPU (Ironwood) 7th generation MERCHANT GPUS NVIDIA	AWS CUSTOM SILICON Trainium3 + Inferentia Training & inference MERCHANT GPUS NVIDIA	Microsoft CUSTOM SILICON Maia 200 Inference accelerator MERCHANT GPUS NVIDIA	Meta CUSTOM SILICON MTIA Ranking & recommendation MERCHANT GPUS NVIDIA
---	--	---	---

The accelerator became a design choice.
No single vendor will own the GPU pod — and the largest buyers are spending real engineering money to keep it that way.

Sources: Google TPU (Ironwood, 7th gen); AWS Trainium3 & Inferentia; Microsoft Maia 200 (announced Jan. 26, 2026); Meta MTIA.

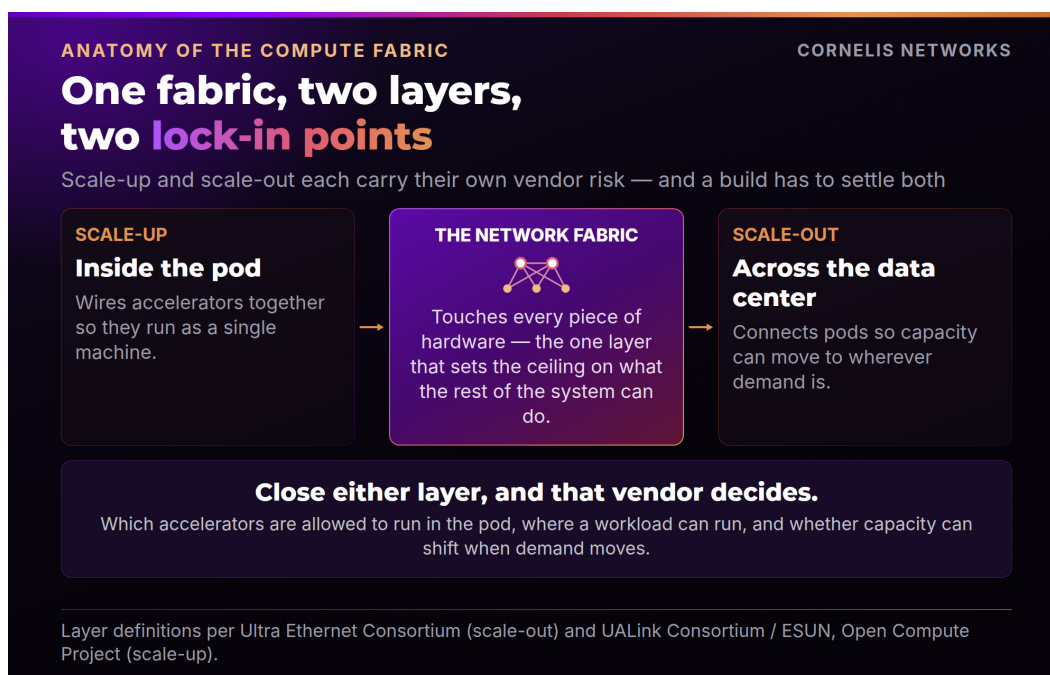


All that hardware comes together at the network. Accelerators from different vendors do not connect to each other directly. They each connect to the network, and the network connects to all of them. That makes it the one layer touching every piece of hardware in a mixed data center, and the layer whose limits set the ceiling on what the rest of the system can do.

The network carries two distinct jobs, and most conversations cover only one.

- Scale-up connects the accelerators inside a pod so they run as a single machine. Scale-out connects pods across the data center.
- A build has to settle both when it picks a network. Leave one unspecified and someone else settles it for you.

A closed network ties your data center to one vendor, and it does so at whichever layer you leave unspecified to them. Close the scale-up layer, and whoever built it decides which accelerators are allowed to run inside the pod. Close the scale-out layer, and whoever built it decides how the cluster expands, where a workload can run, and whether capacity can shift when demand moves. That network lasts longer than any other part of the system, so the commitment outlasts every other decision you make. The network was meant to connect the infrastructure. A closed one quietly begins to govern it.



For a decade, the industry treated the network as a passive pipe: something that moved data between endpoints. That posture was tolerable when compute was the bottleneck and idle time was cheap. Neither condition holds today. Treating the network as passive is now the most expensive assumption in the data center, and the fabric you buy on that assumption is the hardest decision to reverse.

The Current Response

Most operators respond the way any rational buyer would, reaching for one vendor's integrated stack. The appeal is real:

- fastest time to deploy
- lowest integration risk
- one number to call when something breaks.



The problem is what happens to the risk they were trying to avoid. Integration risk does not vanish when the whole stack comes from one source. It converts into exit risk, and that risk gets repriced at every expansion. These choices are sticky, and they carry gravity. The next purchase is easier to make from the same vendor than from anyone else, so the threat of switching turns into a negotiation the buyer is set up to lose. Surrender the alternatives and you pay list price for the life of the deployment.

The damage concentrates at one layer. At scale-out, an operator unhappy with the fabric has an Ethernet ecosystem to defect to. At scale-up, where GPUs are wired together to act as a single system, that exit closes. The dominant interconnect there is proprietary and supports one vendor's GPUs, and NVLink is the clearest example.

The Size of the Gap

When you put a number on the waste, it stops looking abstract. On a cluster of 10,000 GPUs, each percentage point of GPU utilization is worth roughly \$3.5 million a year, about 100 GPUs running flat out at a typical cloud rate.² Even heavily optimized large-scale training systems report model FLOPs utilization in the range of roughly 41% to 55%, so the most capable clusters convert little more than half their compute.³ The most expensive hardware in the data center spends much of its life waiting.



A GPU stuck at half utilization still draws full power and full cooling. The waste lands twice, first in the compute nobody used, then again in the energy and thermal budget spent keeping idle silicon ready. On the balance sheet, that is capital paying rent on itself.

Network lock-in creates four costs over the life of a build:

- **Limited accelerator choice.** The buyer may not be able to use the best processor for each workload.
- **Stranded capacity.** Workloads and capacity may not move freely when demand changes.
- **Longer deployment cycles.** Separate scale-up and scale-out fabrics can add integration and bring-up time.
- **Lower workload efficiency.** Traffic mismatches can reduce throughput and increase cost per token.

Lock-in is not a one-time cost. It is the interoperability and adaptability you gave up on day one, and you continue paying for throughout the life of the deployment.

Openness does not recover that utilization by itself. It preserves the ability to deploy the architecture that does, and to keep deploying it as accelerators turn over.



What to Expect from Your Infrastructure

This is where the conversation shifts from diagnosis to a standards-based requirement. Open standards used to be a preference, favored the way buyers favor good documentation. Now, three forces have turned openness from a philosophy into a requirement. The capital at stake made lock-in a risk the C-suite now owns rather than an engineering annoyance. Credible accelerator alternatives have arrived, making choice real instead of theoretical. At the same time, open specifications have matured to the point where interoperability can be written into an RFI and enforced. Ultra Ethernet addresses scale-out networking, while UALink and ESUN address scale-up.⁴

HOW TO SPEC AN OPEN NETWORK CORNELIS NETWORKS

Change what you require

Four lines to hold firm before a fabric choice becomes permanent

STOP ACCEPTING	START REQUIRING
What a vendor's pitch deck calls "open"	What actually keeps the exit open
✗ A single vendor's roadmap Passed off as the industry spec ✗ One layer covered Scale-up or scale-out named, never both ✗ A proprietary API Locks the application to one network ✗ Paper compliance Spec support with no production record	✓ Cite a named spec Ultra Ethernet, UALink, or ESUN — named body, named spec ✓ Cover both layers Scale-up and scale-out, so the seam can't lock you in ✓ Stay portable Reach the fabric through Libfabric, not a proprietary API ✓ Prove it Proven production performance, at scale, in the field

Requirements per Ultra Ethernet Consortium, UALink Consortium, ESUN (Open Compute Project), and Libfabric (OpenFabrics Interfaces).

An operator writing that requirement into the next build should hold four lines firm:

- **Cite a named spec.** Define openness by referencing an established specification so the fabric itself is something you can buy from more than one vendor. Name the spec and the body that publishes it: Ultra Ethernet (Ultra Ethernet Consortium), UALink (UALink Consortium), or ESUN (Open Compute Project).
- **Cover both layers.** Specify support for the networking layer the workloads require, whether it's scale-up, scale-out, or both, so the boundary between them does not turn into another lock-in point. Open standards across both layers are also what let accelerators get matched to workloads on economics rather than fabric compatibility. The UALink membership, which includes AMD, Apple, AWS, Cisco, Google, HPE, Intel, Meta, and Microsoft, shows where the industry is heading. Nobody funds a standard to escape a vendor they expect to dominate forever.
- **Stay portable.** Require applications to access the fabric through an open interface like Libfabric rather than a proprietary API, making it easier to move workloads across networks without costly rewrites.
- **Prove it.** Prioritize solutions that pair open standards with proven production performance. A strong track record in real-world deployments matters as much as specification compliance.



What Comes Next

The fabric remains the longest bet in the build, and this cycle has raised the stakes on getting it right. Every accelerator decision, every expansion, and every workload placement runs through it.

Open standards have become the lowest-risk path for AI infrastructure, and the move from preference to requirement is already underway. Our job is to make the open path the highest-performing one. At Cornelis, we took a proven architecture and built the hard parts these standards now describe: lossless transport and adaptive routing, proven in the field at national-laboratory scale, including Lawrence Livermore National Laboratory and the Texas Advanced Computing Center. That is the engineering that connects whatever accelerator you run and adapts as fast as your workloads change. Openness has to include the software as well. Libfabric is the open interface AI and HPC applications use to reach the network, and it is what lets teams build software across hardware vendors. We are one of its primary contributors. So when we say open, it is not a commitment we are making. It is work we are already doing publicly, where anyone can verify it.

Standards set the floor. They guarantee that the fabric you buy can be bought from someone else, that accelerators are chosen on economics, and that software moves. What they do not do is set the ceiling. Compliance is where competition starts, not where it ends. The architecture built above that floor decides what the cluster actually delivers.

We are building for both sides of the fabric. Scale-out, connecting pods across the data center, is where we started and where we are proven today. We are now bringing that same architecture and experience to scale-up, connecting the accelerators inside the pod, built on UALink and ESUN. One open architecture across scale-up and scale-out. Cornelis calls this the Active Compute Fabric™: an AI and HPC interconnect that works for the workload rather than only carrying it. Where InfiniBand and Ethernet forward packets between endpoints, it computes at every hop. No passive fabric can be retrofitted into one.

We are building what comes next on those standards, without asking you to give up the openness that got you there. The network you choose determines how much choice you keep. Build it open, and the network becomes the reason you have options, not the reason you don't.

References

1. Custom silicon alongside merchant GPUs, each confirmed on the vendor's own documentation: Google's TPU line is in its seventh generation, [Ironwood / TPU7x](#). AWS builds Trainium for training and inference, now [Trainium3](#), and [Inferentia](#) for inference. Microsoft's Maia is an inference accelerator, [Maia 200 announced January 26, 2026](#). Meta's MTIA serves ranking and recommendation and inference workloads.
2. On a 10,000-GPU cluster, one percentage point of GPU utilization is about 100 GPUs; at a typical cloud rate near \$4 per GPU-hour, that is roughly \$3.5 million a year ($100 \times \$4 \times 8,760 \text{ hours} = \$3.50\text{M}$). Utilization economics discussed in HPCwire, ["From GPU Capacity to Productivity: Closing the Operationalization Gap,"](#) April 23, 2026
3. Model FLOPs Utilization at scale: MegaScale (ByteDance / Peking University) reports 55.2% MFU training a 175B model on 12,288 GPUs, "MegaScale: Scaling Large Language Model Training to More Than 10,000 GPUs" [arXiv:2402.15627](#). Meta's Llama 3 405B reports roughly 41% BF16 MFU on 16,000 H100s, "The Llama 3 Herd of Models" [arXiv:2407.21783](#).
4. Ultra Ethernet Consortium (UEC), scale-out. UALink Consortium, scale-up accelerator interconnect; member roster at [ualinkconsortium.org/members](#). ESUN (Ethernet for Scale-Up Networking), an OCP workstream launched at OCP Global Summit in October 2025, with the ESUN 1.0 base specification released in [month] 2026. Layer mapping and status per Open Compute Project and Synopsys.

Learn more about industry leading AI and HPC scale-out network at [cornelis.com](#)



Other names and brands may be claimed as the property of others. All information provided here is subject to change without notice. Contact your Cornelis Networks representative to obtain the latest Cornelis Networks product specifications and roadmaps. Cornelis Networks technologies' features and benefits depend on system configuration and may require enabled hardware, software or service activation. Copyright © 2026, Cornelis Networks. All rights reserved.