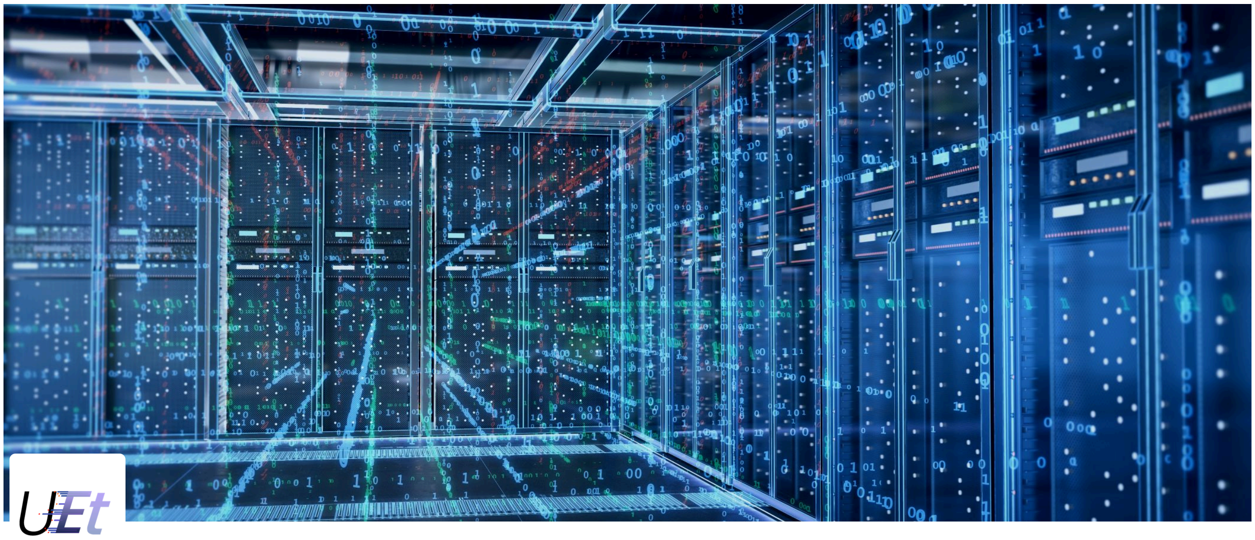


Ultra Ethernet Transport: Accelerate AI with UET

In the race to unlock the full power of AI, data center networks and algorithms are under unprecedented pressure. Properly designing and utilizing the network can be the difference between maximizing GPU performance and leaving expensive hardware idle. The choice of network transport, **RoCE (RDMA over Converged Ethernet)** or **UET (Ultra Ethernet Transport),** and how it's configured, will determine whether networks can deliver on performance, scalability, and efficiency.

Scala Computing's **network simulation platform**, now featuring a newly released **UET NIC simulation model**, gives network engineers the ability to **compare RoCE and UET under identical conditions**, understand the performance tradeoffs, and determine which transport best fits their environment. This improved understanding of the scalability challenges and trade-offs of each protocol enables engineers to make the right decisions that directly accelerate AI algorithm performance and improve GPU utilization.

RoCE has been the workhorse for RDMA traffic in data centers. However, RoCE fundamentally assumes a **lossless network**:

- **Requires PFCs** (Priority Flow Control) to prevent packet loss.

- **Sensitive to out-of-order delivery**, limiting design flexibility.
- **Overhead of lossless enforcement** leads to **underutilized bandwidth** and **deadlock risk**.

At modest scales, these limitations are manageable. But in networks designed for **AI clusters with tens of thousands of GPUs**, the advantages of UET become interesting and should be explored.

Important features of UET

The **Ultra Ethernet Consortium's UET** is engineered for real-world, lossy networks and offers the following important features:

- **Loss tolerance**: No reliance on lossless fabric or PFCs.
- **Out-of-order tolerance**: Flexible routing without strict ordering.
- **Flow spraying**: Multi-path distribution to maximize bandwidth utilization.
- **Minimized tail latency**: Lower end-to-end latency without PFCs.
- **Efficient congestion control (NSCC)**: Handles both **network impairments** and **NIC impairments**.
- **Proven scalability**: Supports networks of **256K nodes today, with a path toward 1M nodes**, without deadlock scenarios.
- **Much lower latency for AI workloads**: Benchmarks (UEC specifications) show significant performance gains in model training throughput.

AI and GPU Utilization

Data-intensive AI training and HPC workloads demand continuous, high-volume data movement. In these environments, network transport is more than a detail, it determines whether GPUs run at maximum capacity or sit idle. Even short stalls or congestion events interrupt the data pipeline, leaving GPUs underutilized or idled and training jobs delayed. This leads to wasted computational cycles and a significant increase in the total time to complete a training job.

RoCEv2 relies on **Priority Flow Control (PFC)** to enforce lossless delivery. While effective in small environments, PFC introduces serious challenges at scale:

- **The PFC Stall:** A single congested flow can pause all traffic on a priority class, leading to **head-of-line blocking** and network-wide stalls.
- **The Utilization Gap:** Even milliseconds of delay cut directly into GPU and network utilization, making training jobs take significantly longer.
- **Deadlocks at Scale:** As networks grow, the probability of **PFC deadlocks** increases exponentially. These deadlocks can freeze large portions of a cluster, requiring complex troubleshooting and costly downtime.

The new Ultra Ethernet Transport (UET) avoids these pitfalls by shifting congestion control into the transport layer, where it can be managed more intelligently:

- **Intelligent Congestion Management:** UET uses retransmits and explicit congestion notification to handle congestion in-band, keeping bandwidth high and GPU pipelines full.
- **Multipath Routing:** By distributing traffic across multiple paths, UET minimizes oversubscription and reduces the likelihood of bottlenecks.
- **Resilience at Scale:** Without reliance on PFC, UET avoids the cascading deadlocks.

Ultra AI Algorithms

For AI researchers and engineers, network simulation with UET isn't just about faster execution of existing models, it opens new frontiers in algorithm design. With UET-enabled simulation, Machine Learning Engineers can:

- **Co-Design Algorithms with the Network:** Test how distributed training algorithms interact with UET's congestion control and multipath routing. Adjust collective operations (e.g., all-reduce, all-to-all) to exploit UET's strengths.
- **Explore Scalability Trade-offs Early:** Understand how algorithms behave at full data center scale, without waiting for those issues to be discovered after deployment.
- **Improve Training Efficiency:** Design algorithms that maximize GPU utilization by aligning data exchange patterns with UET's high-throughput, low-stall characteristics.

- **Lead the Transition:** As the market migrates toward UET, the first algorithms optimized for this new standard will gain a competitive edge, delivering faster training times, lower costs, and better scalability.

The central insight is simple: **high network utilization equals high GPU utilization.** By using Scala Computing's UET NIC simulation to align algorithms with next-generation transport technology, Machine Learning Engineers can deliver AI models that not only run faster today but are positioned to lead in tomorrow's UET-driven data centers.

Simulating UET with Scala Computing

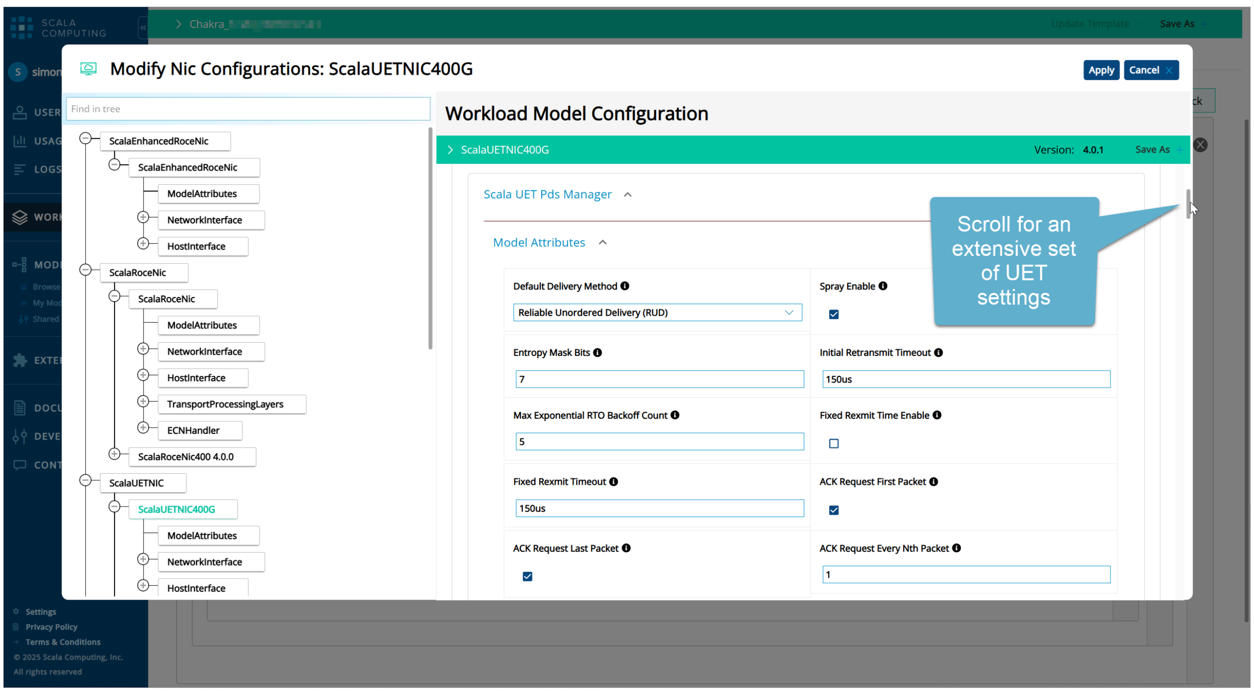
Scala Computing's simulation platform exposes an extensive set of Ultra Ethernet Transport (UET) settings through an intuitive UI, giving engineers the ability to explore system tradeoffs with precision. These controls make it possible to replicate real-world conditions, experiment with protocol behavior, and analyze performance under different configurations, all without the cost of hardware deployment.

Key controls available in the platform for setting up the UET NIC behavior include, but are not limited to:

- **Traffic Classes:** Define and partition bandwidth across eight classes (e.g., 90% data, 10% control).
- **Priority Flow Control (PFC):** Enable or disable PFC per class to test congestion and flow-control strategies.
- **Buffer Sizes & Scheduling:** Adjust egress buffer sizes and scheduling policies to study queuing dynamics.
- **PCIe Interface Modeling:** Simulate PCIe events and inject receiver-side impairments for realistic system-level behavior.
- **Transport Processing Modes:** Configure both Packet Delivery (PD) and Congestion Management (CM) sublayers with detailed transport parameters for Reliable Ordered and Unordered Delivery (ROD or RUD).
- **Telemetry & Packet Capture:** Filter, capture, and analyze packet flows as if an in-line probe were present.

- **Congestion Control (NSCC):** Fully model congestion-control behavior for head-to-head comparison with other transports.
- **Per-Device State:** Manage NIC-level flow control to explore fairness across devices.
- **Bandwidth-Delay Product (BDP):** Tune switch pipeline behavior to optimize responsiveness under load.

By combining these granular controls with high-scale simulation, the platform enables engineers to diagnose bottlenecks, test new strategies, and optimize UET networks with confidence.



Why Simulation Is Essential

Building even a small-scale test environment for UET can be expensive and time-consuming. Deploying hardware, training teams to configure and operate it, and developing the right tools all carry significant CAPEX and OPEX.

By starting with high fidelity simulations, Scala Computing's platform reduces risk by allowing engineers to study and compare RoCE and UET, side-by-side, with the exact same workloads and topologies, or with a variety of different configurations, in order to optimize their network, their AI algorithms or both.

With our simulator, teams can:

- **Compare RoCE and UET head-to-head** under identical workloads and topologies.
- **Test real or synthetic LLM traces** across a single rack or a full data center and compare performance of various AI algorithms.
- **Model impairments** in PCIe and network fabrics to stress-test resilience.
- **Explore parameters** without committing to costly hardware deployment.
- **Validate scale** by simulating tens of thousands of endpoints.
- **Leverage built-in monitoring** for fine-grained transport statistics.

Simulation enables engineers to understand the **network or algorithm tradeoffs** at **data center scale,** before they commit, ensuring design choices that maximize performance, reliability, and cost-effectiveness.

Conclusion

The competitive race to scale with AI and HPC workloads is already underway, and the transport you choose will determine whether your network or algorithm thrives or struggles.

Whether you're validating a next-generation cluster design, troubleshooting bottlenecks, planning a migration path, or improving an AI Algorithm, Scala makes it possible to compare RoCE and UET head-to-head with real workloads, explore tradeoffs under realistic conditions, and design with clarity before you commit resources.

Scala lets you Test UET vs RoCE at data center scale without the cost, risk, or downtime of doing it in hardware so you can predict and optimize your network or algorithm performance with confidence before a single cable hits your floor.

See for yourself how UET reshapes performance at scale.

👉 Contact Scala Computing today for a demo, or visit our website to learn more.