SOLUTION BRIEF

Keysight AI Data Center Builder

AI / ML infrastructure evaluation and benchmarking

KEYSIGHT

Introduction

Benchmarking artificial intelligence (AI) and machine learning (ML) cluster fabric with realistic workloads typically requires significant investment in compute systems, such as graphics processing units (GPUs) and remote direct memory access network interface cards (RDMA NICs). These systems are not only expensive but also time-consuming to build and operate. The challenge lies not just in the financial outlay but also in the complexity of maintaining such sophisticated infrastructure.

This technical overview introduces how Keysight AI (KAI) Data Center Builder addresses these challenges by accelerating AI / ML networks and input/output innovation cycles. This innovative software uses high-density traffic load appliances that support 100-800GE ports to emulate AI hosts and realistic AI / ML traffic patterns. You will learn how KAI Data Center Builder can help speed up releases, lower costs, simplify setup and maintenance, and improve overall performance.

Challenges

To address the challenges for benchmarking AI and ML cluster fabric with realistic workloads, a comprehensive solution is necessary to minimize both cost and complexity. Here are five key components of such a solution:

Realistic emulation: The solution reproduces high-scale AI workloads with measurable fidelity, providing deep insights into collective communication performance.

Simplified benchmarking: The product includes prepackaged benchmark applications, making it easier to validate AI network fabric.

Flexible test engines: Users can choose between hardware load appliances, software endpoints, or real AI accelerators to compare benchmarking results.

Automated test methodologies: These methodologies help qualify AI network fabric efficiency in terms of job completion time, performance isolation, load balancing, and congestion control mechanisms.

Solution: KAI Data Center Builder

The Keysight AI Data Center Builder (KAI DC Builder) is a robust evaluation and benchmarking solution that optimizes AI / ML system design. This solution is faster to deploy and operate, providing deeper insights and higher precision of per-flow measurements. Figure 1 illustrates the various levels within a data center where KAI DC Builder simulates workload behavior.

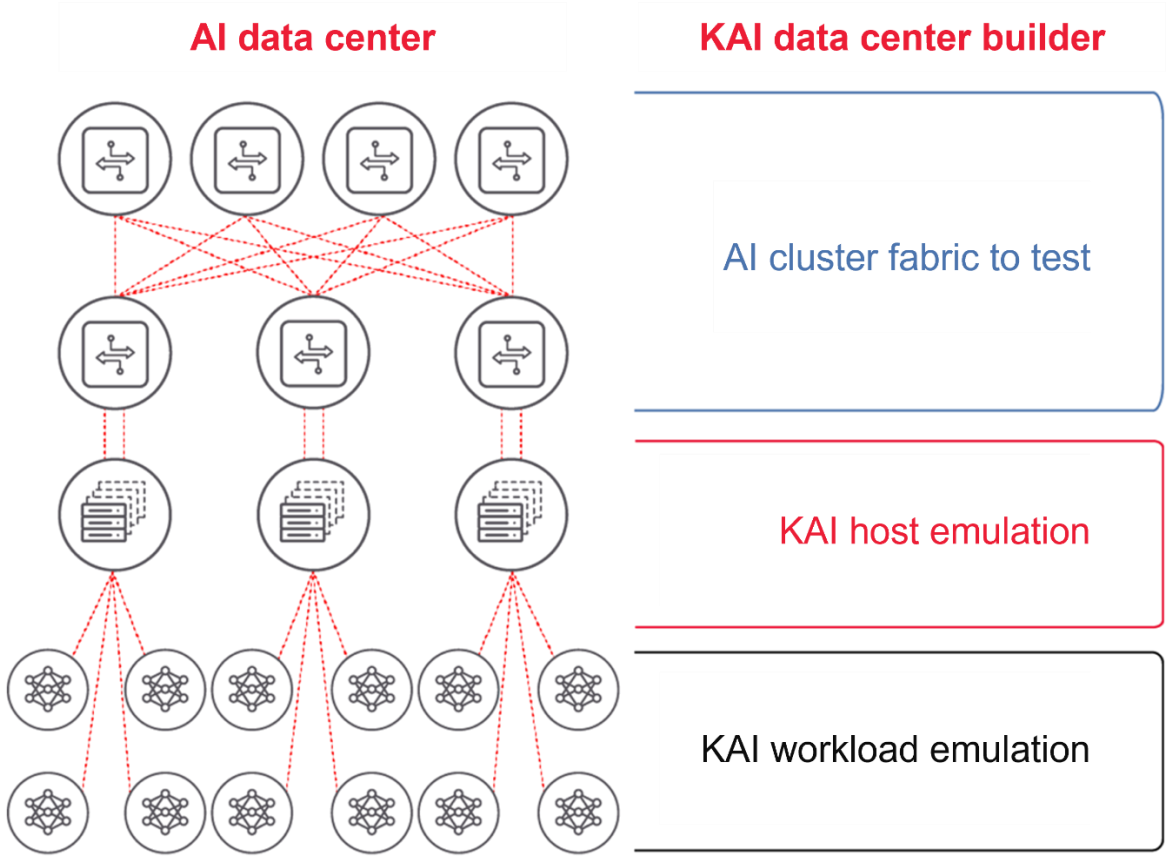


Figure 1. The KAI DC Builder operates on multiple levels to reproduce AI workload behavior

The KAI DC Builder enables you to host and run applications that specialize in measuring or analyzing various aspects of the AI infrastructure. This capability provides detailed insights into performance, helps identify bottlenecks, and optimizes the overall efficiency of your AI systems.

The role of the applications is to collect a dataset by executing a specific trial. For example, one application might focus on measuring data transfer bandwidth, while another might analyze fabric utilization. Each application defines the trial's objective, initializes it with the application-specific trial configuration, and reports results based on the collected dataset. You can interact with the KAI DC Builder applications via the web-based user interface for visual, user-driven workflow or use automation scripts for batch execution from an unattended continuous integration pipeline.

Figure 2 illustrates how KAI DC Builder has two applications that can significantly enhance your AI infrastructure's performance and efficiency:

KAI Collective Benchmarks runs performance micro-benchmarking for distributed communications algorithms commonly used in scaled-out AI systems called collective communications. The typical objective of using the Collective Benchmarks app is to make sure the network delivers the optimal and consistent bandwidth for a range of data exchanges expected from the distributed AI jobs.

KAI Workload Emulation replays short sequences of steps that real AI workloads, like pre- or post-training, go through thousands and millions of iterations called epochs. These workloads form a mix of computational and communication steps, mostly defined by the type of AI model partitioning schemas to distribute the job across hundreds or thousands of GPUs. The Workload Emulation app provides a way to experiment with the performance of the model partitioning choices over various types of network topologies. It helps to find an optimal combination that would minimize the time spent on moving the data and prioritize computations on GPUs, ultimately reducing the Job Completion Time (JCT).

The combination of these two applications results in a more cohesive and automated testing environment, reducing the complexity and time required for setup and analysis. Users gain more accurate insights and can optimize their AI systems more effectively.

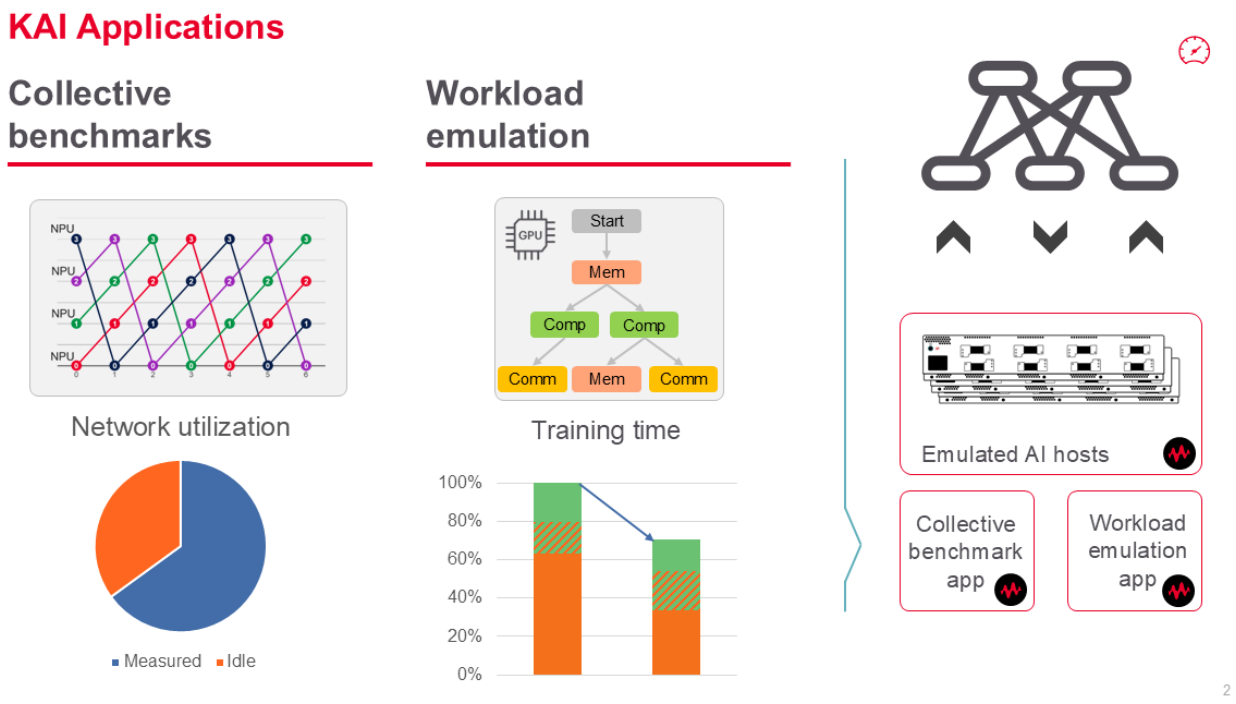


Figure 2. KAI Data Center Builder applications

KAI Test Engines

The KAI applications can execute a trial run using one of the KAI test engines that include:

Keysight AresONE hardware manages high-density traffic loads with RoCEv2 emulation to accurately replicate AI/ML communication patterns at scale over a test fabric. RoCEv2, a transport protocol for Remote Direct Memory Access (RDMA) over Ethernet, is utilized by AI cluster nodes for collective communications.

Keysight endpoint software operates on general-purpose servers equipped with RDMA NICs but without the GPUs to include the network cards and their configuration as part of the system under test.

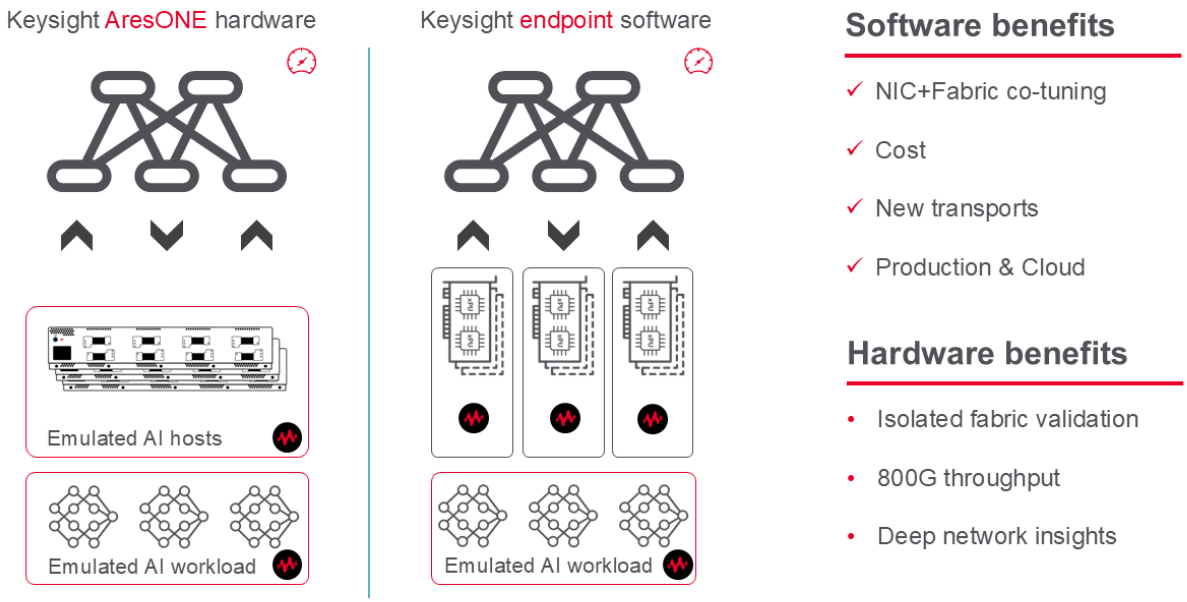
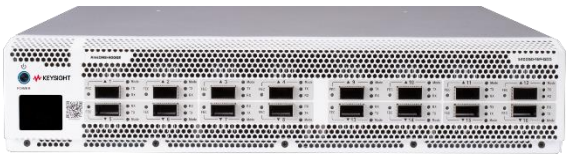


Figure 3. KAI Test Engines choices – AresONE hardware appliances and software endpoints



AresONE-S 400GE

AresONE-M 800GE

KAI Collective Benchmarks

The Keysight Collective Benchmark application can run micro-benchmarking for typical AI communications algorithms on the user-provided AI network fabric, as shown in Figure 4. You can measure fabric performance and make design improvements in a lab or staging environment without connecting AI compute nodes with GPU accelerators to the fabric under test by using RoCEv2 traffic emulation with AresONE hardware.

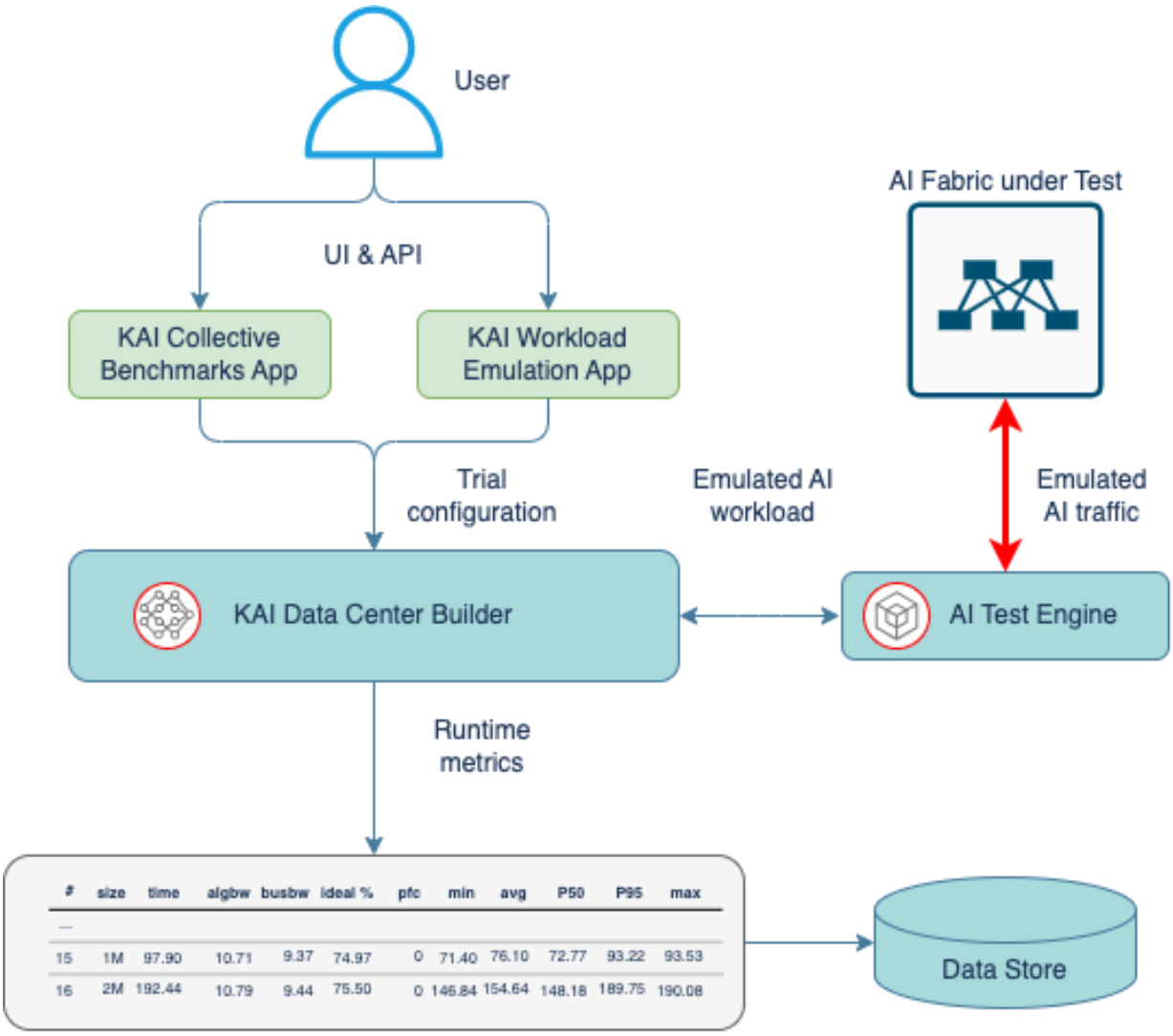


Figure 4. KAI Collective Benchmarks application workflow with AresONE hardware

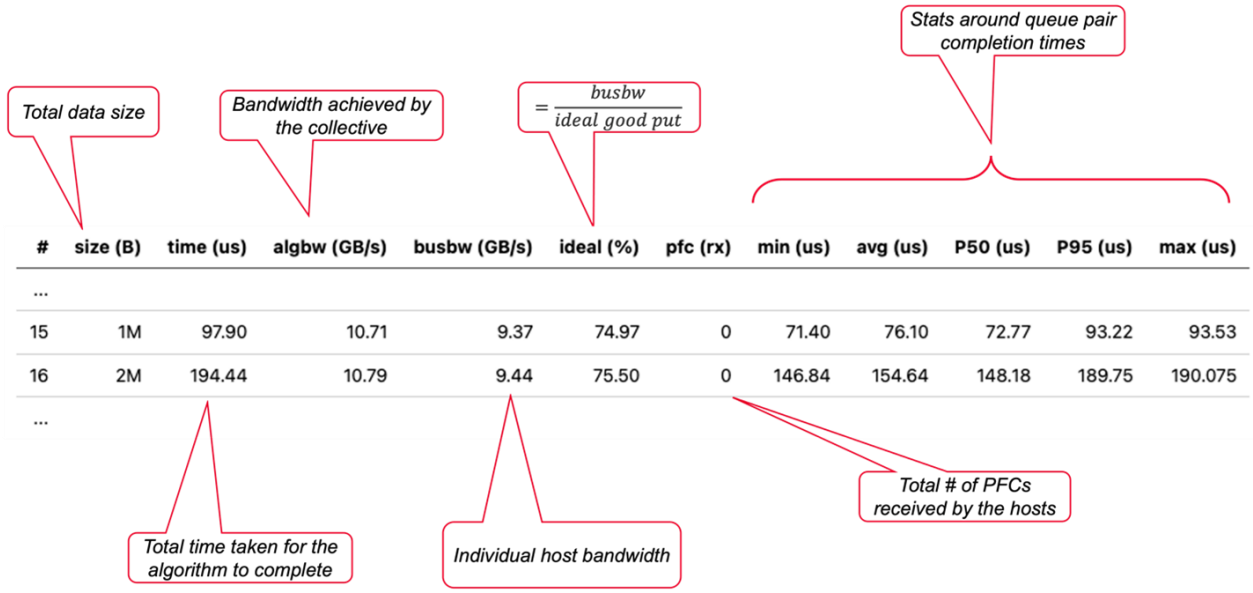
Configuration Parameters

You can parameterize the *trial* configuration for KAI Collective Benchmarks to reflect the size and properties of the AI collective on which to run the benchmark and the AI workload profile that the traffic load appliances will emulate.

Parameter	Values	Description
Logical cluster configuration	• Number of GPUs • GPUs per compute node • NPU interconnect	Describes the AI system configuration for workload emulation
Collective operation: type	• AlltoAll • AllReduce • AllGather • ReduceScatter • Gather • Broadcast	Type of collective operation
Collective operation: algorithm	• Parallel • Unidirectional Ring • Bidirectional Ring • Halving-Double	Collective operation algorithms: applicability of the algorithms depends on the operation type
Workload preset: data size	• Initial value • Iteration step value • End value	Range of data sizes to run benchmarking on
Workload preset: transport profile	• Transport protocol • Number of Q-pairs per rank pair • RDMA message size	Characteristics of transport profile; used to perform fine-tuning on Q-Pair level
Infrastructure preset: NIC profile	• NIC interface speed • NIC MTU • NIC IP interface settings • Traffic class / DSCP value • PFC parameters • DCQCN parameters	Configuration of the congestion control and traffic classification for the emulated NICs

Measurement Metrics

The KAI Collective Benchmarks application reports measurements in a format commonly adopted among AI practitioners. Based on specific parameters of the AI collective and workload, it collects a metrics dataset and presents them in a tabular format. Each row represents measurements for a particular data size the collective needs to exchange. As illustrated in Figure 5, by iterating through a configurable range of data sizes, the KAI Collective Benchmarks application benchmarks the AI fabric performance using a combination of key performance indicators (KPIs) specific for collective communications, like completion time, algorithm, and bus bandwidth, with additional insights into statistical metrics distribution among individual RoCEv2 queue pairs (Q Pairs).



#	size (B)	time (us)	algbw (GB/s)	busbw (GB/s)	ideal (%)	pfc (rx)	min (us)	avg (us)	P50 (us)	P95 (us)	max (us)
...											
15	1M	97.90	10.71	9.37	74.97	0	71.40	76.10	72.77	93.22	93.53
16	2M	194.44	10.79	9.44	75.50	0	146.84	154.64	148.18	189.75	190.075
...											

Figure 5. Measurement metrics example

Trial Summary

	Data Size	Collective	Completion Time (ms)	Algbw (GB/s)	Busbw (GB/s)	Ideal (%)	Min FCT (ms)	P50 FCT (ms)	P95 FCT (ms)	Max FCT (ms)
0	16.00 MB	ALL_REDUCE-1	6.58	2.55	4.46	9.10	0.05	0.05	0.05	0.05
1	32.00 MB	ALL_REDUCE-2	7.20	4.66	8.16	16.65	0.09	0.09	0.09	0.09
2	64.00 MB	ALL_REDUCE-3	11.44	5.87	10.27	20.95	0.17	0.18	0.29	0.33
3	128.00 MB	ALL_REDUCE-4	10.99	12.21	21.37	43.60	0.35	0.35	0.39	0.40
4	256.00 MB	ALL_REDUCE-5	16.51	16.26	28.45	58.04	0.69	0.70	0.95	1.12
5	512.00 MB	ALL_REDUCE-6	26.16	20.52	35.91	73.27	1.39	1.40	1.62	1.85
6	1,024.00 MB	ALL_REDUCE-7	46.03	23.33	40.83	83.29	2.77	2.79	3.03	3.16
7	2,048.00 MB	ALL_REDUCE-8	84.70	25.36	44.37	90.52	5.54	5.57	5.81	6.00
8	4,096.00 MB	ALL_REDUCE-9	163.36	26.29	46.01	93.86	11.08	11.16	11.42	11.55
9	8,192.00 MB	ALL_REDUCE-10	318.55	26.97	47.19	96.27	22.15	22.29	22.58	22.85

Figure 6. Example of a KAI Collective Benchmarks trial summary report

KAI Workload Emulation

AI operators use various parallelism strategies, also known as model partitioning, to accelerate AI model training. Practical demonstrations show that aligning partitioning with AI cluster topology and configuration produces exceptional training performance. During the AI cluster design phase, critical questions are best answered through experimentation. For example, many questions address the efficiency of data movement between graphical processing units (GPUs), such as:

- Scale-up design of GPU interconnects inside an AI host or a rack.
- Scale-out network design, including bandwidth per GPU and topology.
- Configure the network load balancing and congestion control.
- Tune the training framework parameters to optimize performance and efficiency.

The KAI Workload Emulation application reproduces network communication patterns of real-world AI training jobs. Its objectives are to accelerate experimentation, reduce the learning curve required to become proficient and provide deeper insights into observed performance degradation causes. These insights would not be easily attainable if a real AI training job were used for experimentation.

Running KAI Workload Emulation in KAI DC Builder enables you to:

- Experiment with model partitioning parameters and rank scheduling over the AI infrastructure.
- Understand the contribution of communications within and among partitions to the overall job completion time (JCT).
- Identify types of collective operations demonstrating low performance and drill down to identify bottlenecks.
- Employ network utilization, tail latency, congestion, and its impact on JCT.

Workload library

KAI DC Builder includes a library of AI workloads, from image classification to large language models (LLMs) like Llama, as shown in Figure 8. It offers a selection of popular model partitioning schemas, including data parallel (DP), fully sharded data parallel (FSDP), and three-dimensional (3D) parallelism. You can import your workload execution traces into the library using MLCommons Chakra format.

Figure 7 illustrates the workloads in the library curated by Keysight, which has detailed metadata documenting the environment and key hyperparameters used to capture their execution traces. The library provides capabilities to explore the behavior of the workloads and understand the types and timing of collective operations, as well as computations.

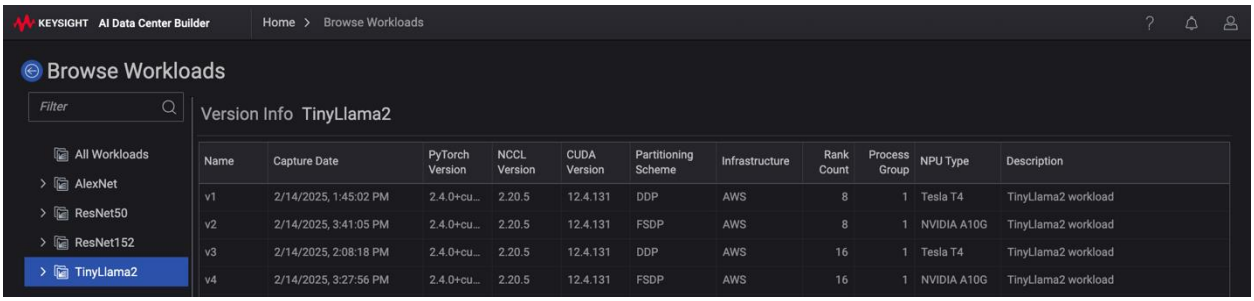


Figure 7. KAI Data Center Builder workload library

Each workload can contain traces collected from each individual rank (NPU) that capture steps the job went through on that compute node, as shown in Figure 8. Except for the simplest cases like DDP, the behavior of different ranks may differ depending on their position in the partition.

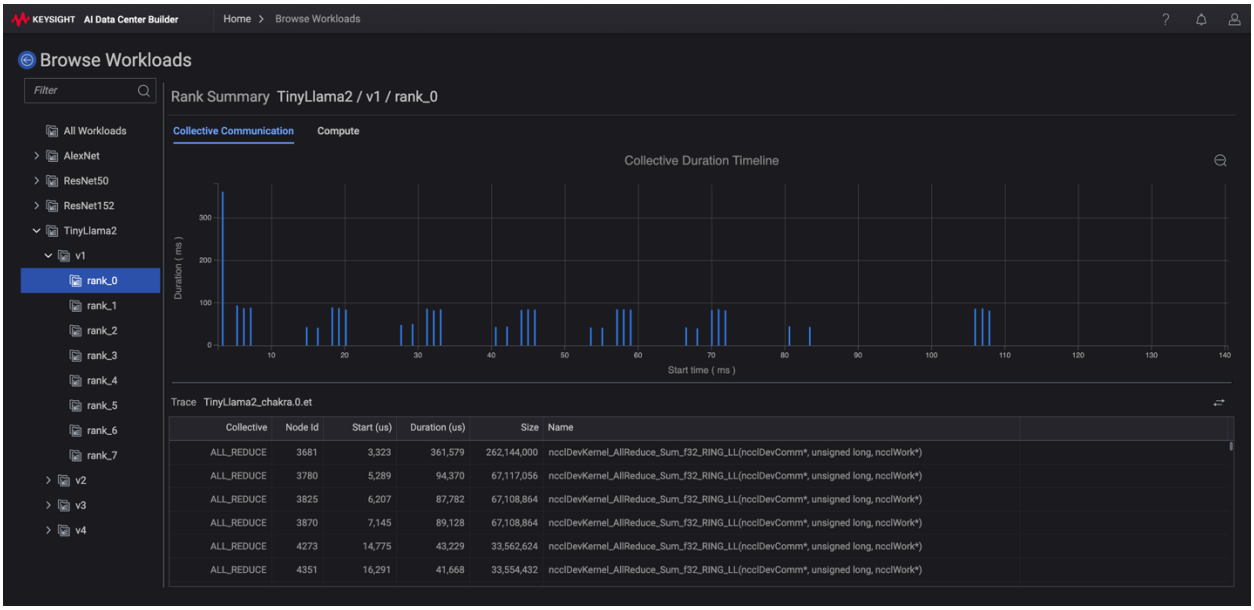


Figure 8. Workload details for the collective duration timeline

Configuration Parameters

After selecting workload traces from the library, a user assigns them to a set of emulated NPUs (ranks) for replay. Like the KAI Collective Benchmarking application, there is a section on the configuration that defines the logical cluster configuration by specifying the number of AI hosts, the number of NPUs per host, and the presence of NPU interconnects. Your test bed must have enough emulation resources (hardware test ports or RDMA NICs) to support the number of elements defined in the logical infrastructure section, as depicted in Figures 9, 10, and 11.

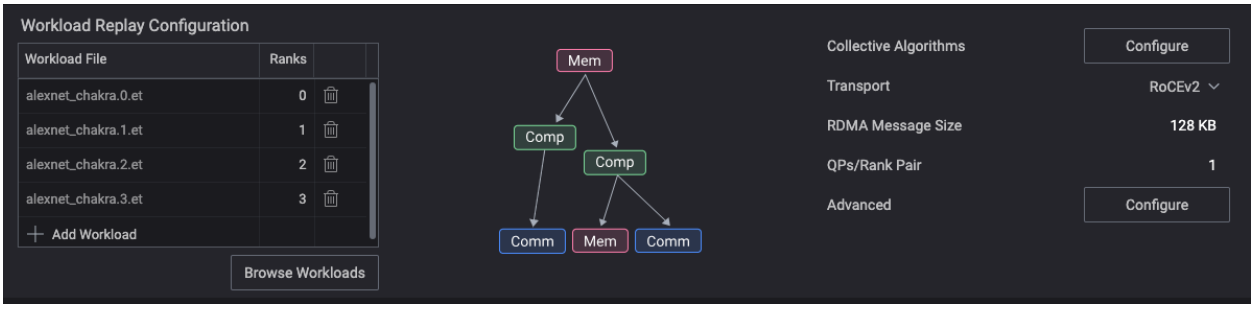


Figure 9. Scheduling of workloads

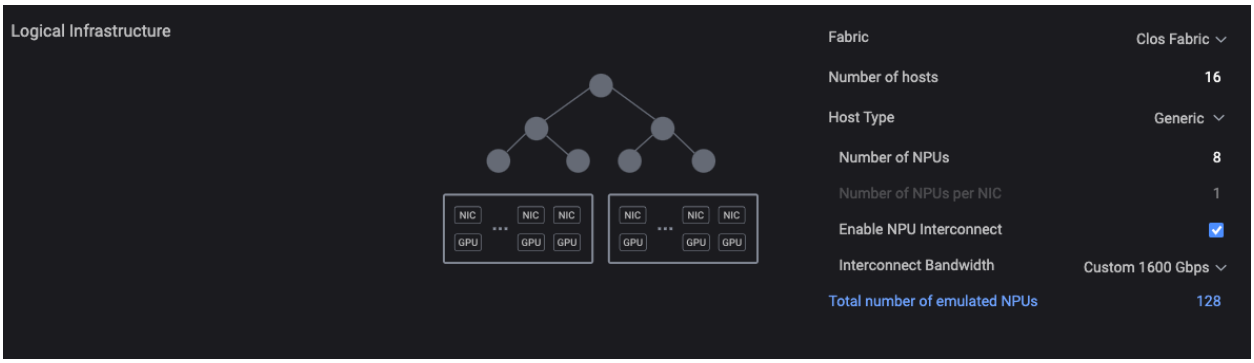


Figure 10. Modeling AI infrastructure with eight racks,16 AI hosts, and eight GPUs

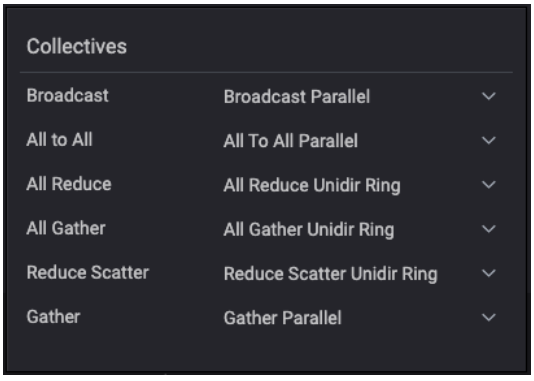


Figure 11. Collective operation parameterization

Measurement Metrics

After running the Workload Emulation trial, the application reports the overall completion time for every workload trace, totaling the time for all the ranks it was scheduled on. Additionally, it provides the performance of each collective operation for all the workloads in the trial, including details such as operation characteristics, completion time, relative start time, and information about the model partition of the operation, as shown in Table 1.

Table 1. Workload emulation measurement summary

#	Workload ID	Collective ID	Data Size (GB)	CCT (s)	Ranks	Start (s)	PP Group	TP Group	DP Group
1	LLM_3DP_1	AllGather_1	0.125	0.32	8	3.56	-	1	-
2	LLM_3DP_1	AllGather_2	0.125	0.29	8	4.12	-	2	-
3	LLM_3DP_1	AllReduce_1	2.0	1.34	16	5.43	-	-	1

Summary

The Keysight AI Data Center Builder delivers a combination of high-fidelity AI workload emulation, prepackaged benchmarking applications and dataset analysis tools to significantly improve the benchmark performance of the AI / ML cluster network fabric.

To accelerate AI / ML network design, the Keysight AI Data Center Builder:

- **Emulates high-scale AI workloads with measurable fidelity —** Gain deep insights into collective communication performance.
- **Simplifies the benchmarking process —** Validate AI network fabric with prepackaged benchmark applications built through partnerships with the largest AI operators and AI infrastructure vendors.
- **Executes defined AI/ML behavior models —** Enable sharing between users and customers to help reproduce experiments.
- **Offers a choice of test engines —** Choose between RoCEv2 endpoint emulation on high-density AresONE traffic load appliances and software endpoints on real AI accelerators to compare benchmarking results.

The KAI DC Builder enables large-scale validation and experimentation with fabric design using the AresONE-S 400GE and AresONE-M 800GE test hardware realistically and cost-effectively. This solution complements the use of GPUs to test AI / ML workloads, enabling AI operators to reduce the spend they would have allocated entirely to GPU-based benchmarking systems for the more scalable, robust, and integrated Keysight AI Data Center Builder.

Learn More

Read the Keysight AI Fabric Test Solution datasheet for details about supported RoCEv2 traffic emulation and congestion control parameters and measurements.

Discover more about: KAI Data Center Builder and Keysight AI solutions.

Have a question? Talk to our product experts.