SNIA DNSF | DATA, NETWORKING, & STORAGE

Storage Requirements for AI

Training and Checkpointing

John Cardente

Technical Staff, Dell Storage CTO Group

The AI boom is driving incredible demand for GPUs leading to a need to maximize their utilization

GPUs Essential for AI

- Modern deep learning AI models require millions of matrix operations
- Matrix operations must be parallelized to make AI ***computationally feasible***
- GPUs designed to do parallel matrix operations quickly and cost effectively.
- GPUs needed to make AI ***economically feasible***

GPUs Expense and Scarce

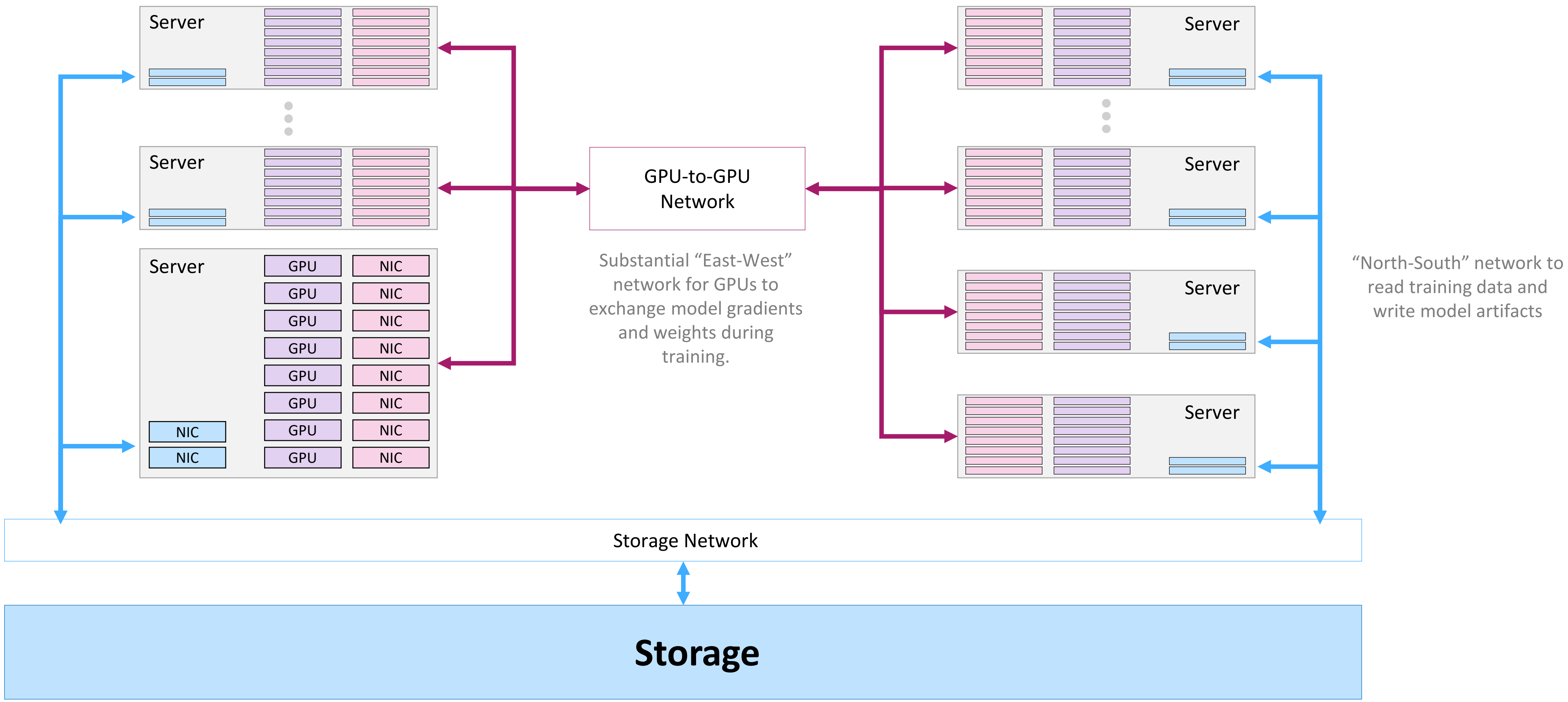
- Companies are racing to build AI datacenters
- AI datacenters can contain 100’s to 1000’s of GPUs
- Demand for GPUs is surpassing supply
- GPUs are becoming ***costly*** and ***difficult to acquire***

Maximizing GPU Utilization Essential

- Demand, cost, and scarcity making GPUs the most valuable AI datacenter asset
- Companies must maximize the use of the GPUs they have
- ***Maximizing GPU utilization*** becoming the main AI datacenter design goal

SNIA DNSF | DATA, NETWORKING, & STORAGE

Maximizing GPU utilization requires balancing compute, network, and storage performance

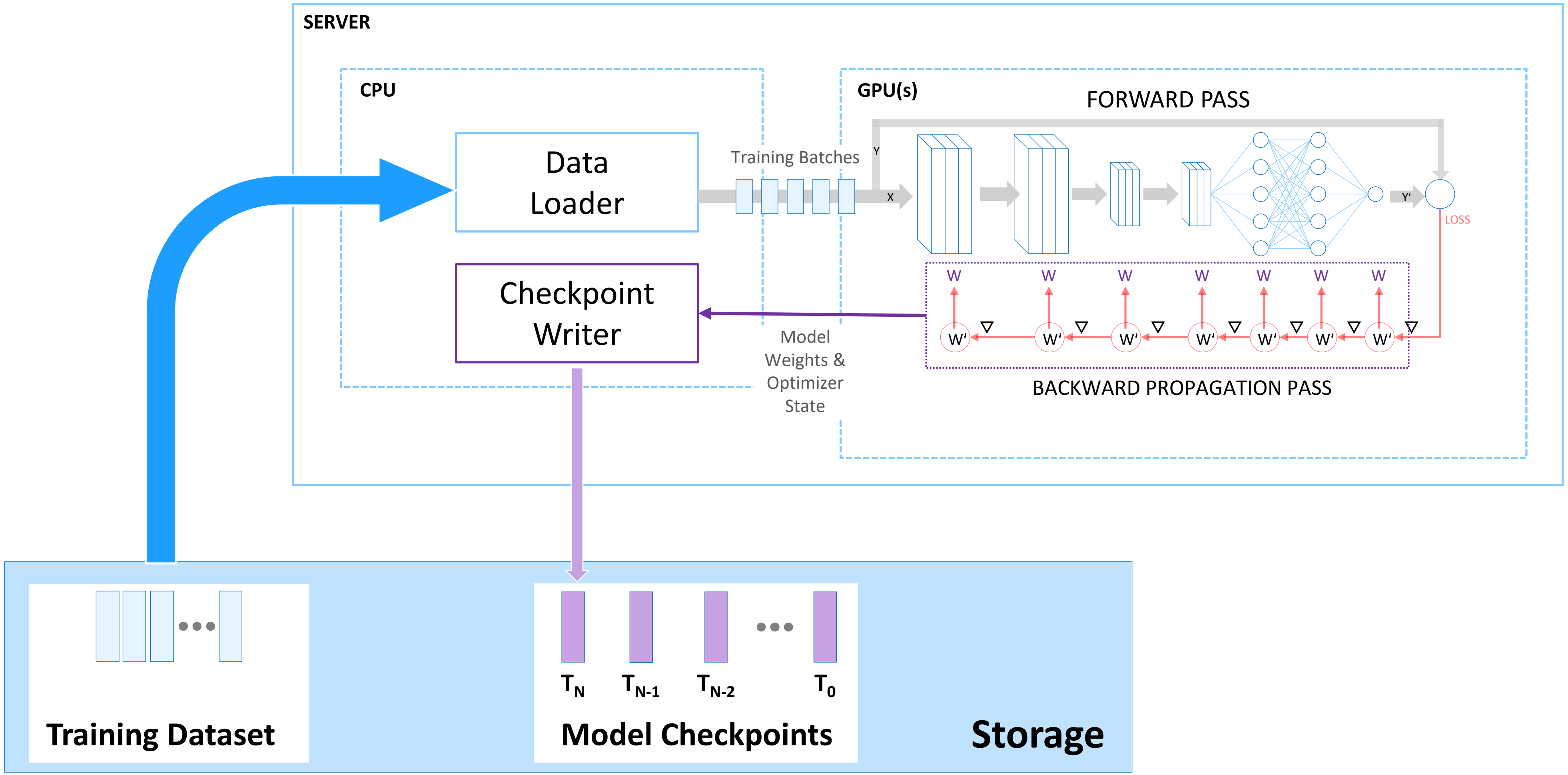


Storage plays an important role across entire AI lifecycle

	Data Preparation	Training & Tuning	Inference
Key Tasks	• Scalable and performant storage to support transforming data for AI use • Protecting valuable raw and derived training data sets	• Providing training data to keep expensive GPUs fully utilized • Saving and restoring model checkpoints to protect training investments	• Safely storing and quickly delivering model artifacts for inference services • Providing data for batch inferencing
Critical Capabilities	• Storing large structured and unstructured datasets in many formats • Scaling under the pressure of map-reduce like distributed processing often used for transforming data for AI • Support for file and object access protocols to ease integration	• Sustaining read bandwidths necessary to keep training GPU resources busy • Minimizing time to save checkpoint data to limit training pauses • Scaling to meet demands of data parallel training in large clusters	• Reliably storing expensive to produce model artifact data • Minimizing model artifact read latency for quick inference deployment • Sustaining read bandwidths necessary to keep inference GPU resources busy

Focus for this talk

AI models trained using batches of training data to update model weights with periodic checkpoints for recovery



Limited information exists on training read performance requirements; GPU benchmarks provide a way to estimate

1. Run AI training benchmarks designed to saturate GPU utilization
2. Extract throughput results in terms of training examples per second
3. Determine size of training examples in bytes
4. Multiply training example throughput and size to estimate training data read bandwidth needed to keep GPUs fully utilized

- MLCommons MLPerf Training benchmark suite ideal for this purpose
- Covers a variety of AI models
- Designed to saturate GPU utilization
- Results from multiple submitters publicly available

Training data storage read bandwidth requirements vary greatly; depends on model compute boundness and input size

Name	Model Size (Parameters)	Training Dataset	Input Size (Bytes)	# H100 GPUs	Model Throughput (Training Examples/Sec)	Training Data Read Bandwidth
Resnet-50 Image Classification	23M	ImageNet LSVRC2012 dataset 1.2M images, **~155GB**	116K	8	55K	**6.1 GB/sec**
BERT-Large Transformer LM	345M	Wikipedia 2020/01/01 dataset, pre-processed to **~86GB**	2K	8	5.2K	0.009 GB/sec
GPT3 LLM	175B	C4 dataset, **~305GB**	8K	32	19.5K	**0.146 GB/sec**
DLRM-DCNv2 Recommender	24B	Criteo 1TB click-through dataset, synthetically extended, ~**3.5TB**	912	8	12M	10.6 GB/sec
3D U-Net 3D Image Segmentation	19M	KiTS19 Kidney Tumor data set 300 3D images, **~27GB**	92M	8	463	**41.6 GB/sec**
MaskRCNN Object Detection	44M	COCO data set, 121K images, **~18GB**	160K	8	1.3K	0.2 GB/sec

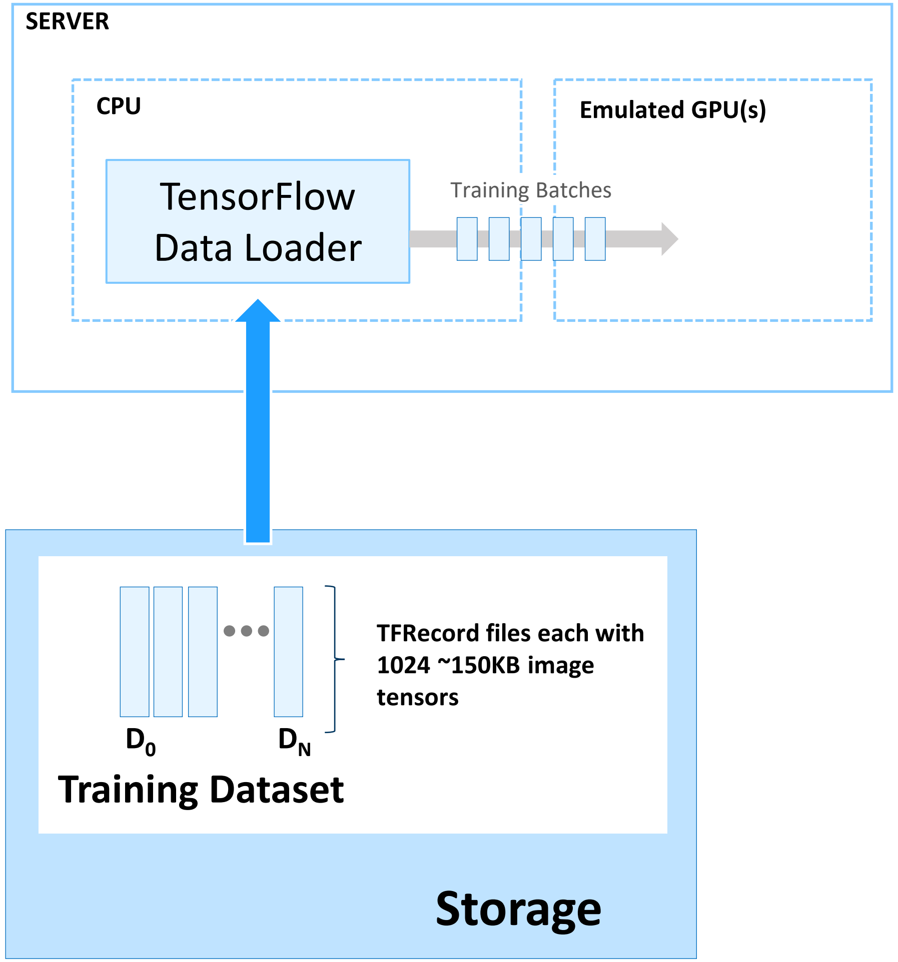
Single instance of GPT3 model required 32 GPUs

Most of DLRM's parameters are embedding tables, less compute bound than size suggests

Based on MLPerf Training v3.0 results for H100 80GB GPUs from multiple submissions

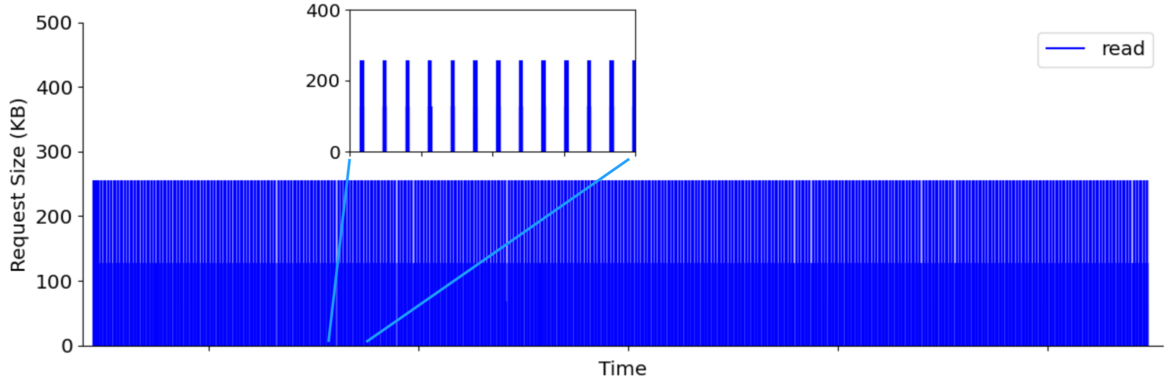
SNIA DNSF | DATA, NETWORKING, & STORAGE

Example: Resnet50 training emulated by DLIO benchmark

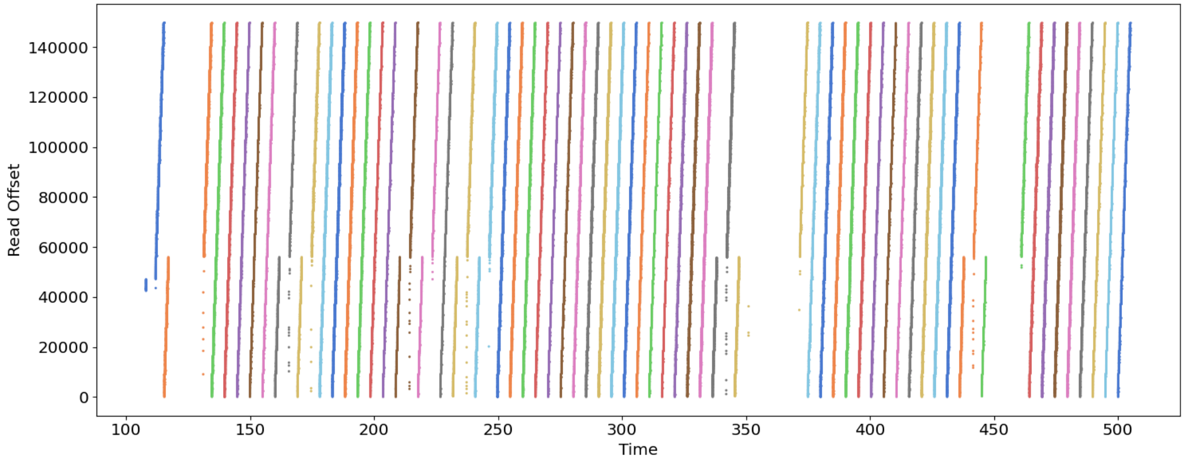


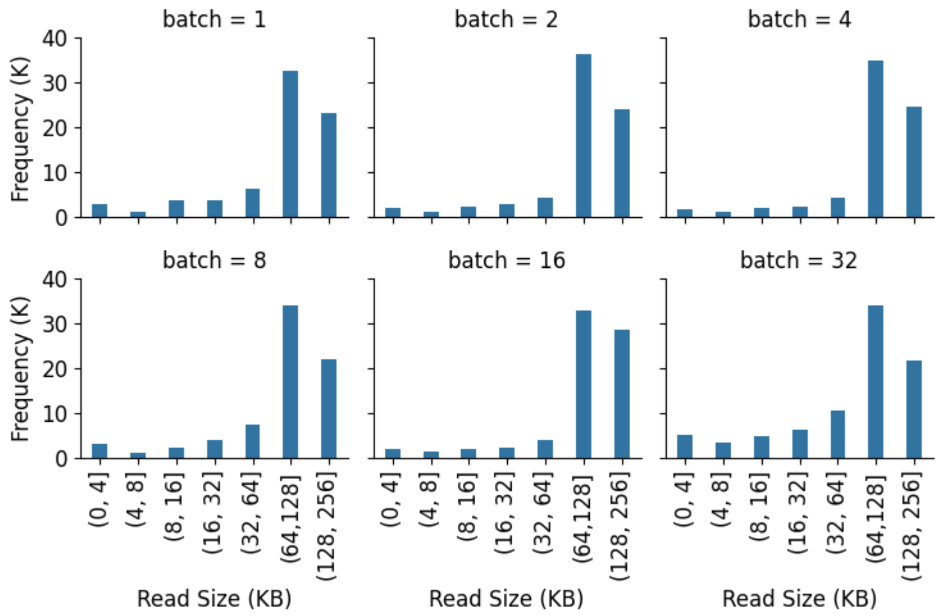
- DLIO modifies TensorFlow Resnet50 implementation to emulate GPUs with a "compute time" delay
- Includes a synthetic data generator that creates TFRecord files with 1024 ~150KB image tensors
- Uses real TensorFlow Data Loader library to access training data
- Produces IO traffic representative of real Resnet50 training

Reading training data via NFS generates a steady stream of sequential 64KB-256KB IOs regardless of batch size



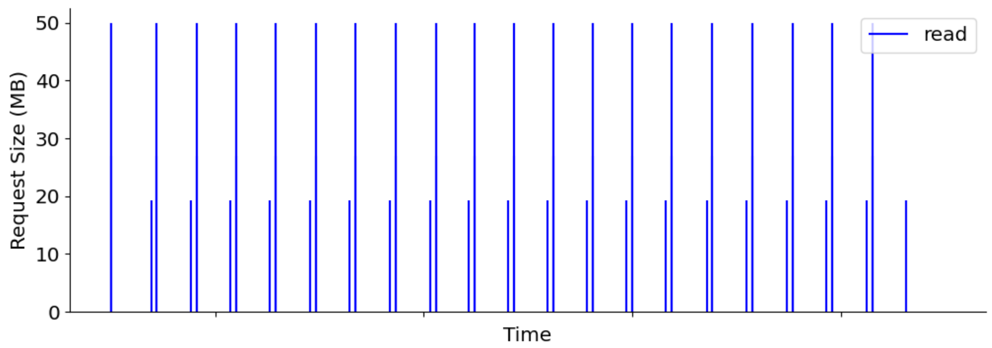
Each color represents a different TFRecord file



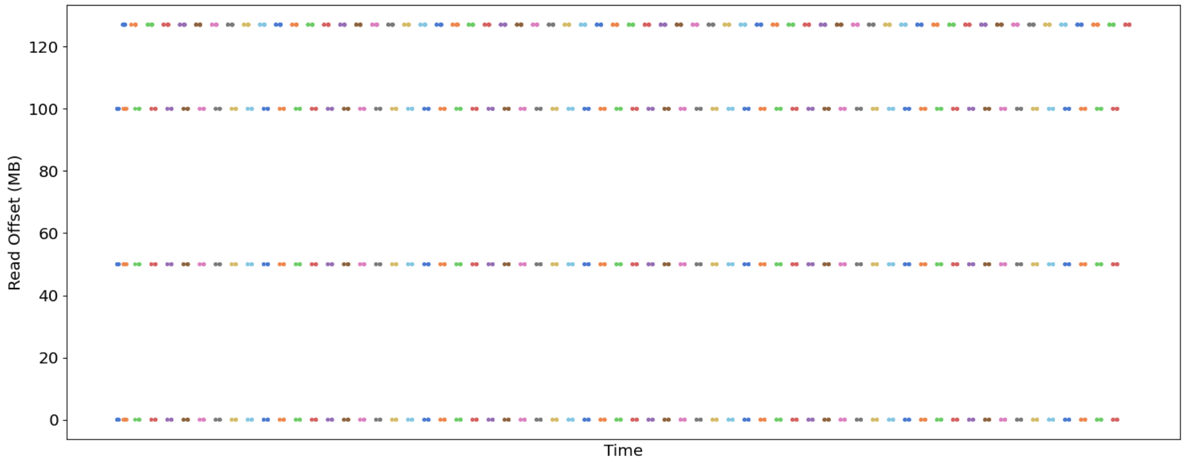


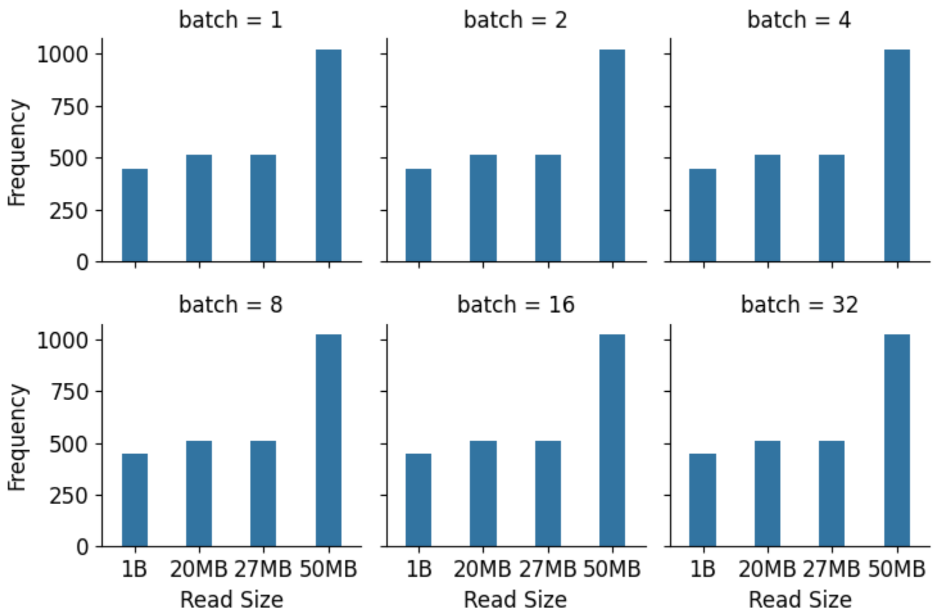
SNIA DNSF | DATA, NETWORKING, & STORAGE

Reading same Resnet50 training data via S3 results in larger sequential 20-50MB IOs



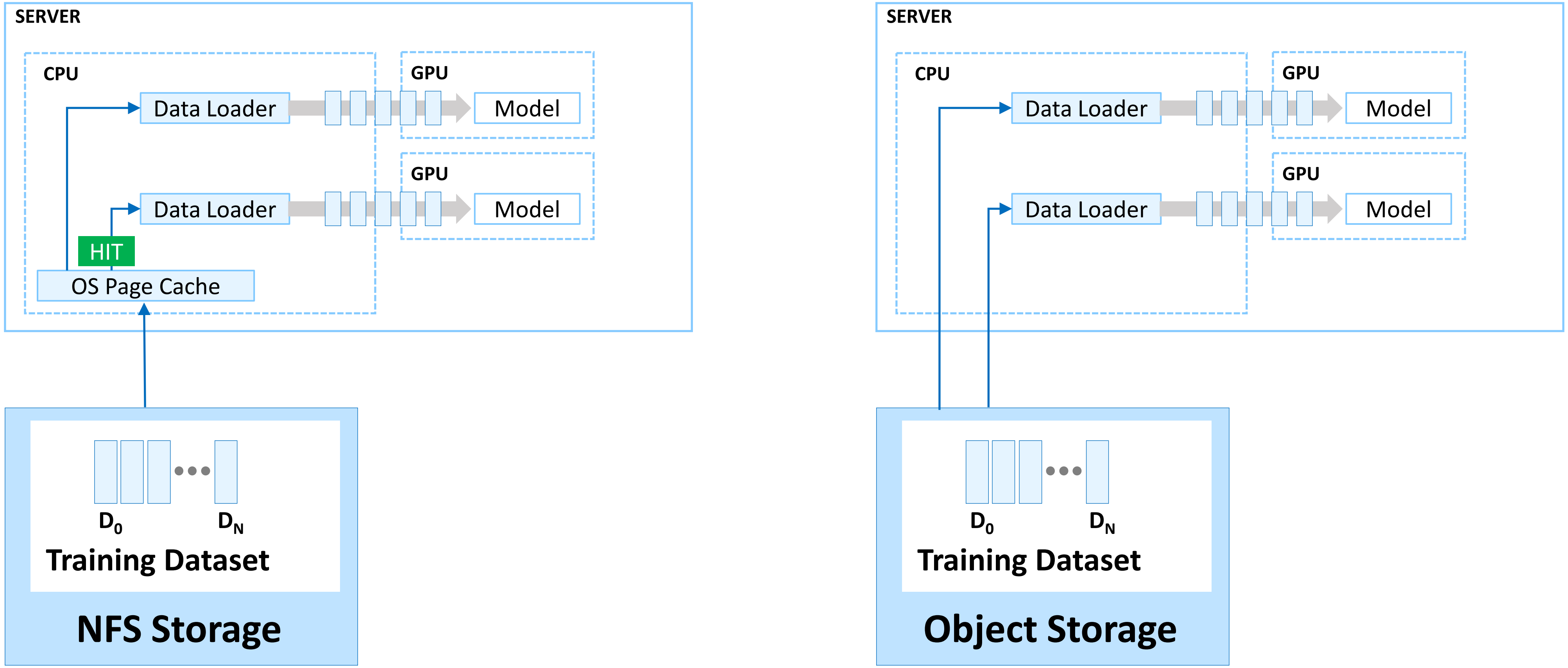
Each color represents a different TFRecord file



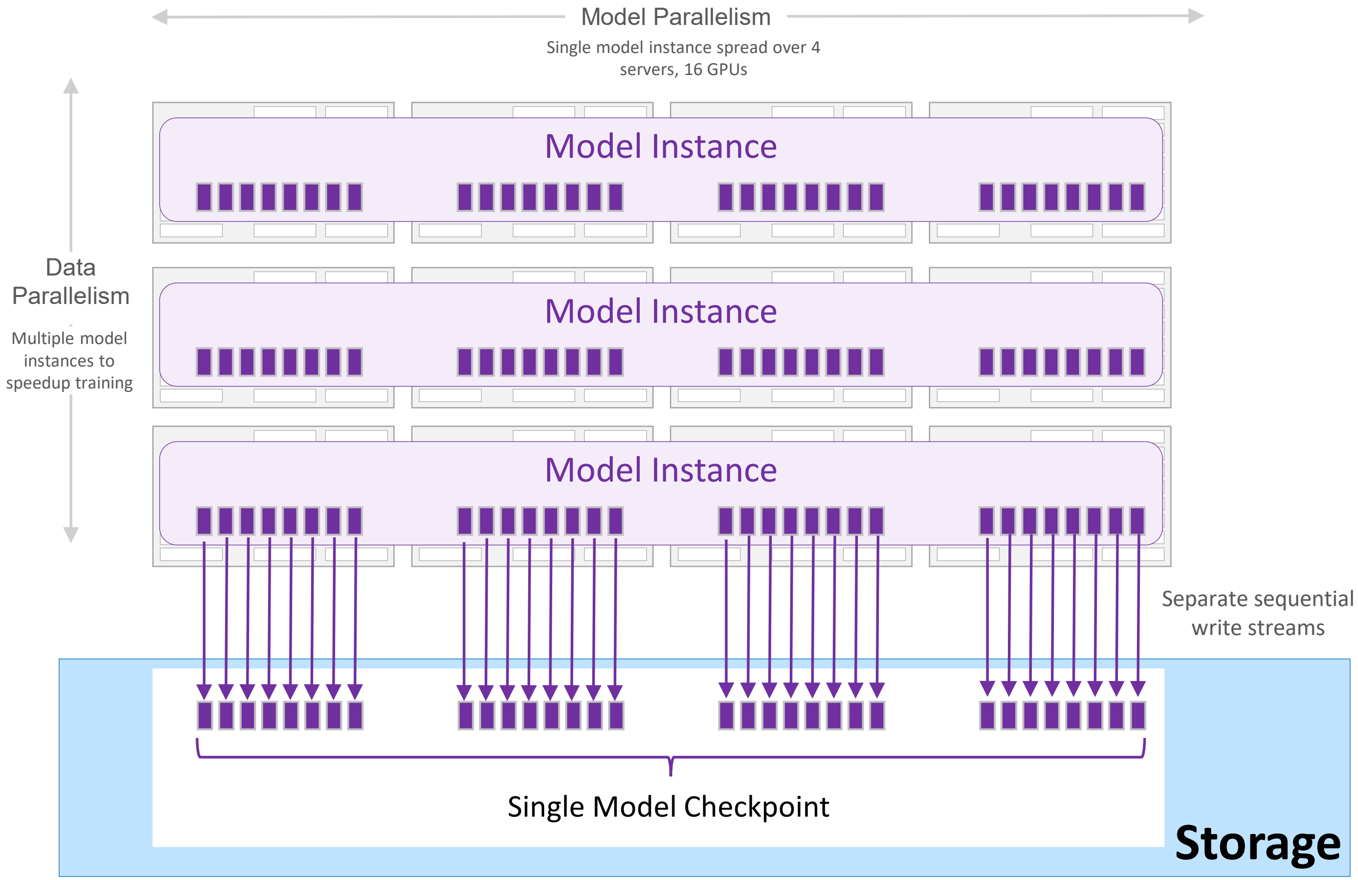


SNIA DNSF | DATA, NETWORKING, & STORAGE

NFS benefits from OS page cache when multiple model instances trained on single server



Checkpoints periodically save training state needed to resume training after a failure or interruption



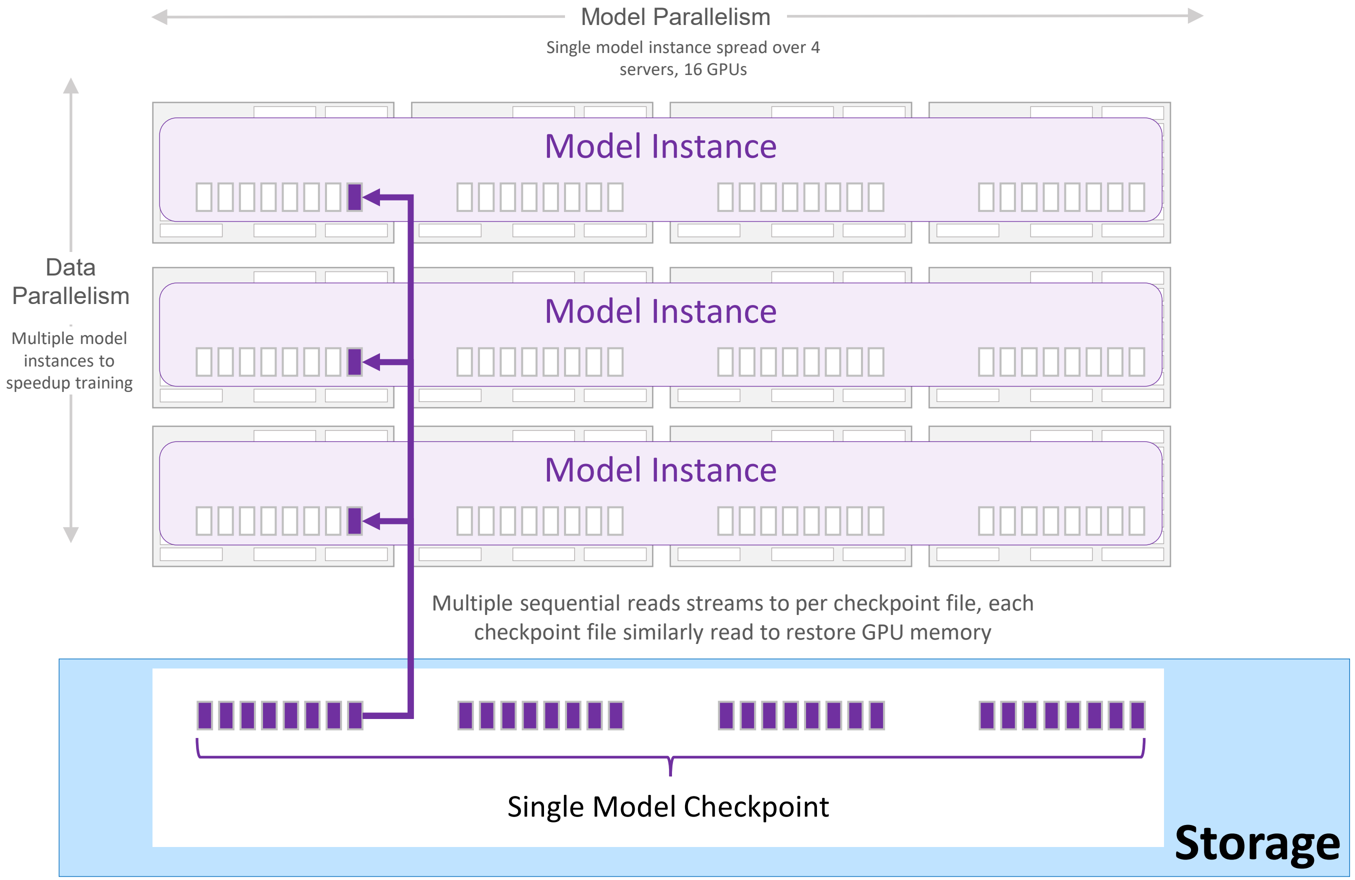
- Checkpoints contain learned model weights and optimizer state information
- Checkpoints may be saved as one or more files; depends on model parallelism and implementation
- Each checkpoint file is sequentially written by one writer
- For data parallelism, only one model instance needs to be saved; don't need to save all of GPU memory
- **Training typically paused during checkpointing reducing GPU utilization, important to complete quickly**

Checkpoint aggregate write bandwidth requirements depend on model size and maximum allowed time

Model Parameters (B)	Checkpoint Size (GB)	Total Write BW (GBps) Needed to Checkpoint within Time Limit 72 sec (1% 2Hrs)	180 sec (2.5% 2Hrs)	360 sec (5% 2Hrs)	540 sec (7.5% 2Hrs)	720 sec (10% 2Hrs)
3	42	0.6	0.2	0.1	0.1	0.1
7	98	1.4	0.5	0.3	0.2	0.1
13	182	2.5	1.0	0.5	0.3	0.3
33	462	6.4	2.6	1.3	0.9	0.6
70	980	13.6	5.4	2.7	1.8	1.4
140	1960	27.2	10.9	5.4	3.6	2.7
175	2450	34.0	13.6	**6.8**	4.5	3.4
530	7420	103.1	41.2	20.6	13.7	10.3

Assumptions:
- Checkpoints every 2 hours
- 2 bytes per model parameter (BF16)
- 12 bytes per model parameter for optimizer and other state

Resuming from a checkpoint requires restoring saved state to all GPUs



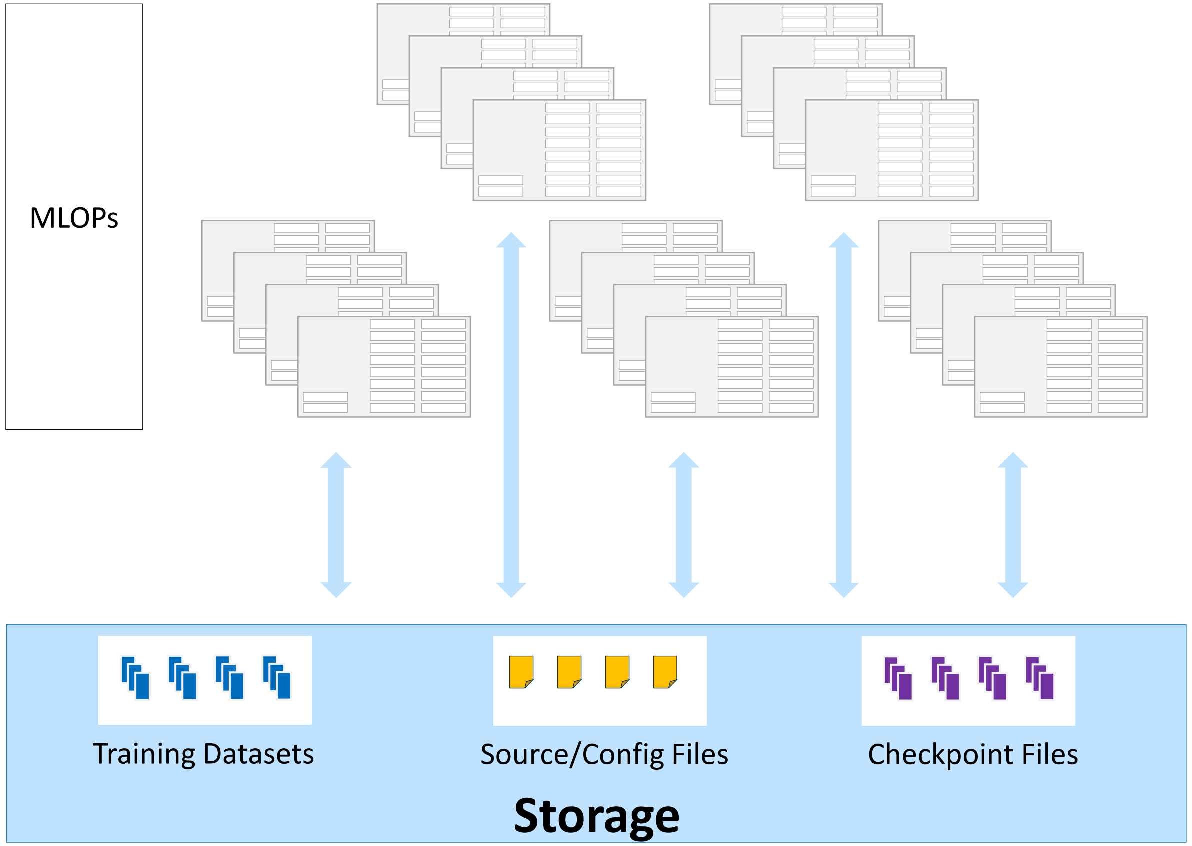
- Each GPU's memory must be re-initialized with weights and optimizer state from appropriate checkpoint file(s)
- Checkpoint files typically read sequentially
- When using model parallelism, a single checkpoint file may be used to restore multiple GPUs
- Number of readers per checkpoint file depends on degree of data parallelism
- **Training unable to start until all GPU memory restored**

SNIA DNSF | DATA, NETWORKING, & STORAGE

Checkpoint aggregate read BW depends on model size, data parallelism, and maximum allowed time

Model Parameters (B)	Checkpoint Size (GB)	Total Read BW (GBps) Needed to Restore Checkpoint within 5 Minutes					
		# Model Instances (Data Parallelism)					
		1	8	16	32	64	128
3	42	0.002	0.02	0.04	0.07	0.15	0.30
7	98	0.01	0.04	0.09	0.17	0.35	0.70
13	182	0.01	0.08	0.16	0.32	0.65	1.29
33	462	0.03	0.21	0.41	0.82	1.64	3.29
70	980	0.05	0.44	0.87	1.74	3.48	6.97
140	1960	0.11	0.87	1.74	3.48	6.97	13.94
175	2450	0.14	1.09	**2.18**	4.36	8.71	17.42
530	7420	0.41	3.30	6.60	13.19	26.38	52.76

GPU clusters run multiple workloads, rely on equal access to data, and require scalable storage performance & capacity



- Modern GPU clusters may contain thousands of servers and 10's of thousands of GPUs
- MLOPs platforms with distributed scheduling used to assign and execute jobs across cluster
- Jobs need access to training, checkpoint, and other data regardless of server deployed on
- Likely many simultaneous storage workloads including data prep, training, and checkpointing
- GPU clusters expected to grow as business demands increase, storage must scale accordingly

AI storage required to perform and scale across AI lifecycle

	Requirements & Considerations	
Reading Training Data	• Accommodate wide range of read BW requirements and IO access patterns across different AI models • Deliver large amounts of read BW to single GPU servers for most demanding models	• Use high performance, all-flash storage to meet needs • Leverage RDMA capable storage protocols, when possible, for most demanding requirements
Saving Checkpoints	• Provide large sequential write bandwidth for quickly saving checkpoints • Handle multiple large sequential write streams to separate files, especially in same directory	• Understand checkpoint implementation details and behaviors for expected AI workloads • Determine time limits for completing checkpoints
Restoring Checkpoints	• Provide large sequential read bandwidth for quickly restoring checkpoints • Handle multiple large sequential read streams to same checkpoint file	• Understand how often checkpoint restoration will be required • Determine acceptable time limits for restoration
Servicing GPU Clusters	• Meet performance requirements for mixed storage workloads from multiple simultaneous AI jobs • Scale capacity and performance as GPU clusters grow with business needs	• Consider scale-out storage platforms that can increase performance and capacity while providing shared access to data

SNIA DNSF | DATA, NETWORKING, & STORAGE

Also consider traditional enterprise storage requirements still applicable to AI

- Data protection
- High availability
- Compression and deduplication
- At rest encryption
- Multi-protocol data access
- Remote and hybrid cloud replication
- Security and governance
- Long term archival storage
- Data lifecycle management

THANK YOU

John Cardente

John.Cardente@dell.com

SNIA DNSF | DATA, NETWORKING, & STORAGE