PURESTORAGE®

GRAPHCORE

REFERENCE ARCHITECTURE

Scaling AI with Graphcore and Pure Storage

Get faster time-to-insight with Pure Storage FlashBlade and Graphcore IPU-POD64.

Contents

Introduction

Data and artificial intelligence (AI) teams today need simple yet powerful infrastructure to take ideas from experimentation to production rapidly. They need end-to-end infrastructure that provides a performant platform that is easy to set up, but that does not impede the work of data scientists and machine learning (ML) engineers. Graphcore and Pure Storage® have brought intelligent compute and storage together to create a converged infrastructure solution to serve ML workloads of all sizes while maintaining simplicity and performance at any scale.

Graphcore provides machine intelligence compute systems that are built around the intelligence processing unit (IPU), a new processor specifically designed for AI computing needs. The IPU's unique architecture enables teams to explore and deploy entirely new types of workloads, driving the next advances in machine intelligence.

Storage is a crucial component of a deployable data center solution. Choosing and optimally configuring the right storage components is critical for users that want efficient and reliable infrastructure solutions. Pure Storage FlashBlade® provides an all-flash unified fast file and object (UFFO) storage platform for consolidating data across your entire ML-ops pipeline. FlashBlade provides the capacity and performance to scale your most data-hungry training jobs and keeps your IPUs fed.

This document provides an example reference architecture that has been developed in close partnership by Pure Storage and Graphcore to support system deployers in configuring an optimal solution and extract maximum performance and value from their IPU-POD configurations as they build and scale their AI compute capability.

Graphcore IPU-POD64 System Overview

The IPU-POD64 system is Graphcore's reference architecture for scale-out for a powerful and flexible AI infrastructure design for your machine intelligence training and inference workloads.

Innovate at Massive Scale

The core building block of the IPU-POD system, or rack, is the IPU-M2000 shelf. This is the fundamental compute engine for machine intelligence from Graphcore and features four units of the powerful second-generation IPU (the GC200 processor), an accelerator designed from the ground up for AI. An individual IPU-M2000 shelf can deliver up to one petaflops of AI compute, up to 256GB exchange memory, in a slim 1U blade to handle the most demanding of machine intelligence workloads.

Figure 1. Graphcore IPU configurations.

The IPU-M2000 shelf has a flexible, modular design, so you can start with one and scale to thousands. It works as a standalone system; eight can be stacked together or racks of sixteen tightly interconnected IPU-M2000s in IPU-POD64 systems can grow to supercomputing scale thanks to 2.8Tb/s high-bandwidth, near-zero latency IPU-Fabric interconnect architecture built into the box.

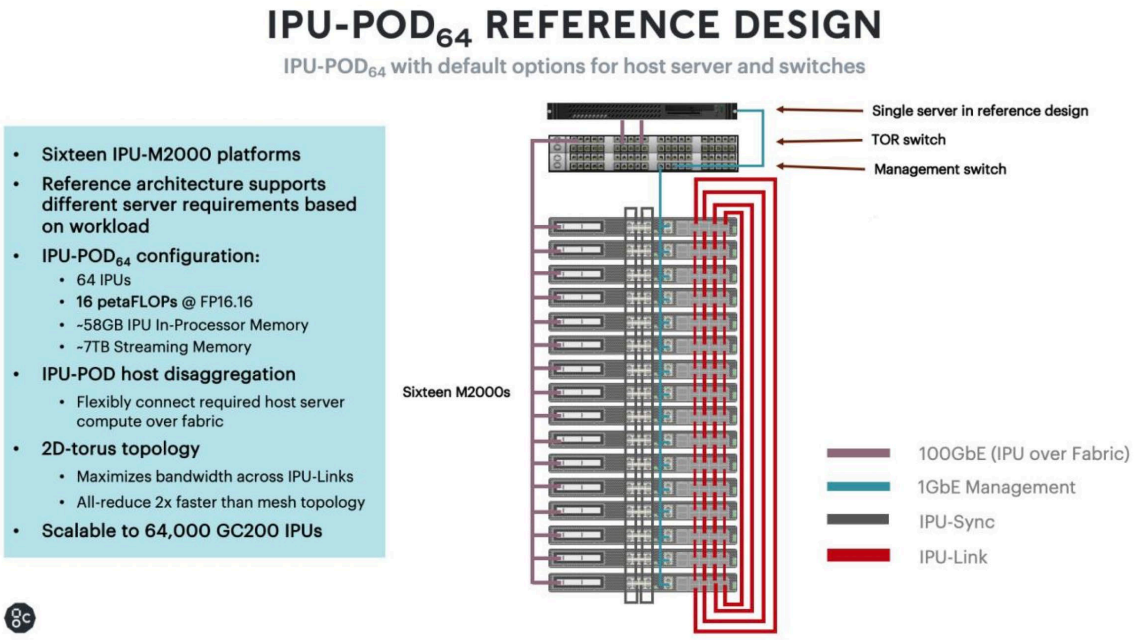

Figure 2. IPU-POD64 reference design.

The IPU-POD64 reference design is a rack solution containing 16 IPU-M2000s, 1 to 4 host servers (the default is 1 host server in the reference configuration), network switches, and IPU-POD software. There are 64 GC200 IPUs in total, with four IPUs in each IPU-M2000.

The IPU-POD64 is designed to deliver 16 petaflops of AI compute in an efficient, flexible, and pre-qualified configuration.

Disaggregated to Scale with Your Needs

AI workloads have different compute demands. For production deployment, optimizing the ratio of AI to host compute can maximize performance and efficiency, and improve the total cost of ownership. IPU-POD64 is a disaggregated system that separates host servers and switches from IPU-M2000 building blocks in a data center. With IPU-POD64, you build a system that is ideally matched to your production AI workload.

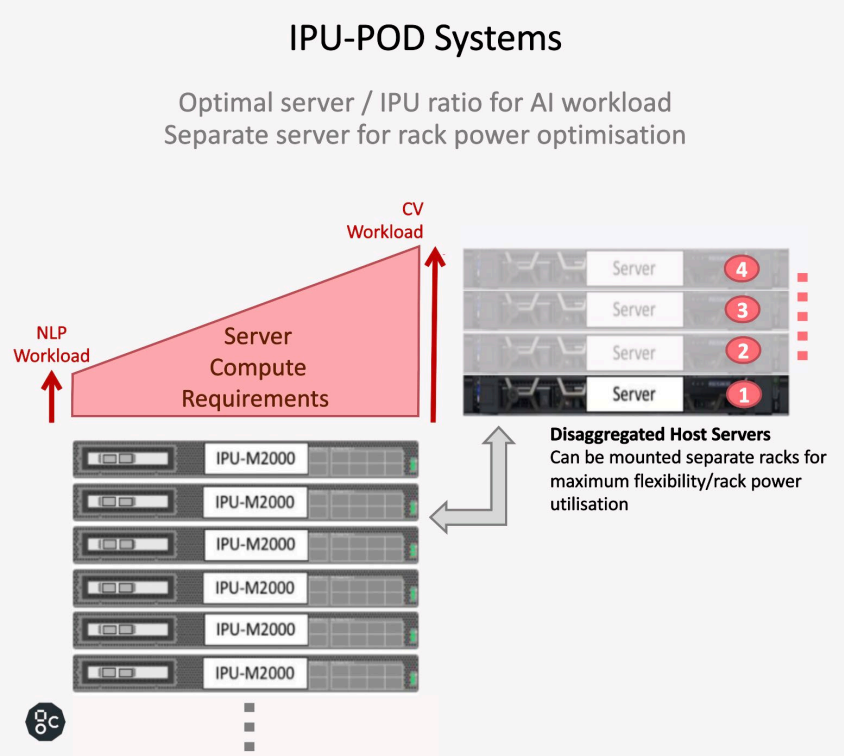

Figure 3. IPU-POD server to IPU ratio for AI workloads.

For example, natural language processing (NLP) models require little CPU interaction and utilization, while convolutional neural networks (CNN)-based workloads such as computer vision (CV) require a larger proportion of scalar computing and benefit from more CPUs being involved. The system can be tailored for the workload.

Unmatched Scale-out with IPU-Fabric

IPU-Fabric is Graphcore's innovative low-latency, all-to-all IPU interconnect. Eliminating communication bottlenecks with reliable deterministic performance is highly efficient, whatever your scale.

Data Center Compatibility

IPU-POD64 brings together powerful IPU compute with a choice of best-in-class data center technologies and systems from leading technology providers in flexible, pre-qualified configurations. This ensures that your data center is operating with maximum efficiency and performance, while making your data center AI deployments simpler and faster. In this paper, Pure Storage FlashBlade is evaluated as a storage solution to support AI workloads running on an IPU-POD64 system.

FlashBlade Overview

Pure Storage developed the FlashBlade architecture to meet the storage needs of data-driven businesses. FlashBlade is an all-flash system, optimized for storing and processing unstructured data. A FlashBlade system can simultaneously host multiple file systems and multi-tenant object stores for thousands of clients.

FlashBlade is a scale-out, all-flash storage system, powered by a distributed file system purpose-built for massive concurrency across all data types. It can scale up to multi-petabyte capacity with linear-scale performance, simply by adding a single blade at a time, up to 150 blades. Due to its native scale-out architecture and ability to drive performance for any type of workload, it is considered a data hub that enables enterprises to consolidate a range of workloads, from backup to analytics and AI, on a single platform.

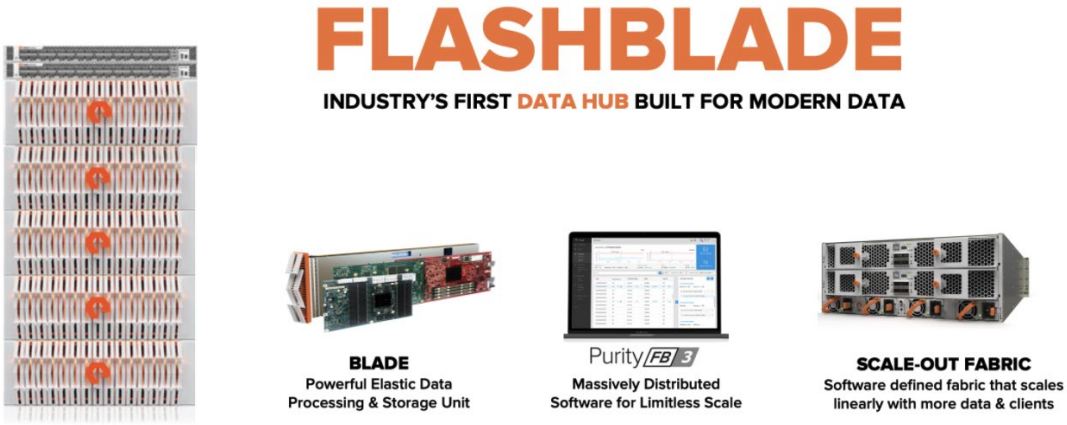

Figure 4. Pure Storage FlashBlade.

A FlashBlade system's ability to scale performance and capacity is based on five key innovations:

- **High-performance storage device:** FlashBlade maximizes the advantages of an all-flash architecture by storing data in all-flash storage units and ditching the crippling, high-latency storage media such as traditional spinning disks and conventional solid-state drives. The integration of scalable NVRAM into each storage unit helps scale performance and capacity proportionally when new blades are added to a system.
- **Unified network:** A FlashBlade system consolidates high communication traffic between clients and internal administrative hosts into a single, reliable, high-performing network that supports both IPv4 and IPv6 client access over Ethernet links up to 1.6Tb/s.
- **Purity//FB storage operating system:** With its symmetrical operating system running on FlashBlade's fabric modules, Purity//FB minimizes workload balancing problems by distributing all client operation requests evenly among the blades on a FlashBlade system.
- **Common media architectural design for files and objects:** The FlashBlade system's single underlying media architecture supports concurrent access to files via a variety of protocols such as NFS, NFS over HTTP, and SMB, as well as object storage via the Amazon S3 protocol, across the entire FlashBlade configuration.
- **Simple usability:** Purity//FB on FlashBlade alleviates system management headaches as it simplifies storage operations by performing routine administrative tasks autonomously. With a robust operating system, FlashBlade is capable of self-tuning and providing system alerts when components fail.

A full FlashBlade system configuration consists of up to 10 self-contained rack-mounted chassis interconnected by high-speed links to two external fabric modules (XFM). At the rear of each chassis are two on-board fabric modules for interconnecting the blades, other chassis, and client systems using TCP/IP over high-speed Ethernet. Both fabric modules are interconnected, and each contains a control processor and Ethernet switch ASIC. For reliability, each chassis is equipped with redundant power supplies and cooling fans.

The front of each chassis holds up to 15 blades for processing data operations and storage. Each blade assembly is a self-contained compute module equipped with processors, communication interfaces, and either 17TB or 52TB of flash memory for persistent data storage. With up to 10 chassis, a single FlashBlade system can scale from the smallest 7 x 17TB (119TB) system to the largest 150 x 52TB (7.8PB) system.

The current FlashBlade system can support millions of NFS metadata operations per second, over 17GiB/sec of 512KiB reads, and over 6GiB/sec of 512KiB overwrites on a 1.5:1 compressible dataset in a single 4U chassis with 15 blades. FlashBlade can scale both compute and performance up to a 10 x 4U chassis with 150 blades.

As mentioned in the previous section, disaggregation is a powerful tool for efficiently scaling infrastructure resources. Storage disaggregation benefits this reference architecture by providing additional dimensions for scaling. With FlashBlade, you can separate storage from the IPU and CPU, allowing you to scale each resource independently. This simplifies finding an optimal balance of resources for any collection of workloads and accelerates response times to changing application needs.

Reference Architecture

This section describes the hosts, storage, and networking configuration used in the IPU-POD64 reference architecture featuring the Pure Storage FlashBlade storage solution.

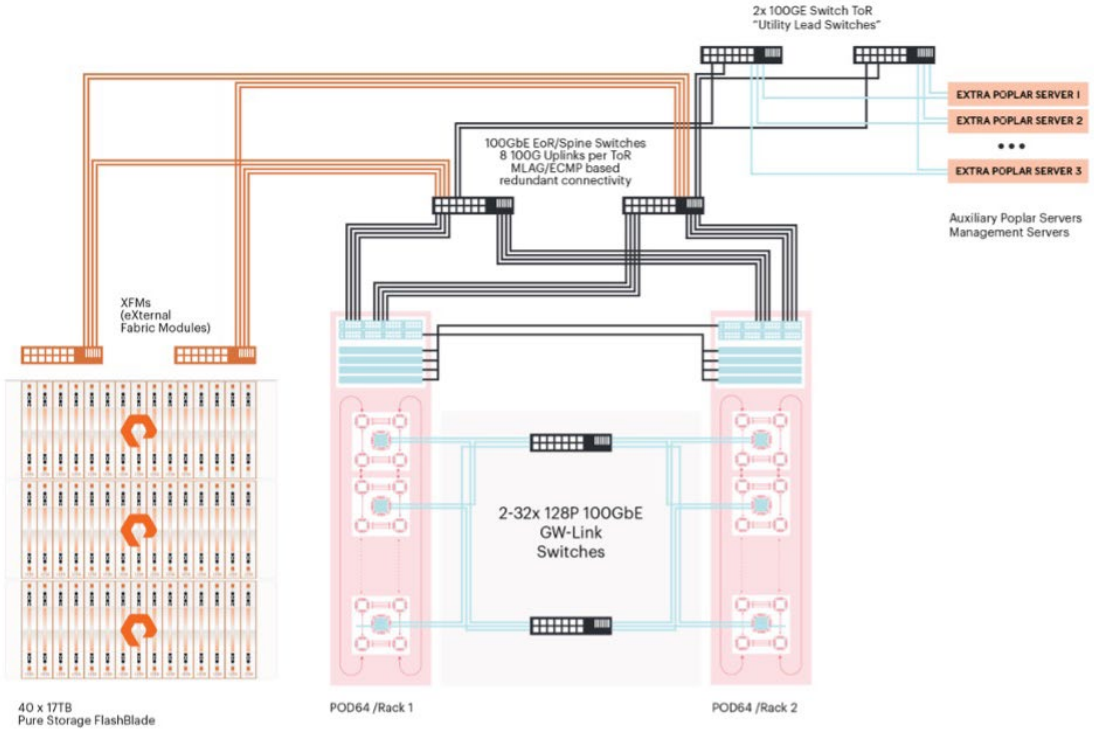

Figure 5. FlashBlade scales linearly from one chassis to 10 chassis, and seven blades to 150 blades.

Poplar Hosts

Poplar is the Graphcore software framework (tools and libraries) for programming the IPUs. Poplar enables the programmer to write a single program that defines both the graph to be executed on the IPU devices and the controlling code that runs on the host. The device code is compiled and loaded onto the IPUs ready for execution.

The Poplar framework is used to manage the host processor and to coordinate and drive the accelerators in the IPU-POD64 instance.

The Poplar host consists of:

- Hardware: Four Dell R6525 AMD EPYC, 512GB RAM, Mellanox connectX-5.
- Software: Ubuntu 18.04 LTS 5.3 (or later) kernel

Network Connections

The network configuration tested and recommended for the IPU-POD64 system is as follows:

- POD 1G management network using Arista DCS-7010T-48-F
- POD Top-Of-Rack for 100G connectivity: Arista DCX-7060CX-32S-ES-F
- Connected via 8-way 100G LAG to: Arista 7060PX4-32-F2 as SPINE 400G

Storage

To support the Graphcore IPU-POD64 system, the FlashBlade storage solution was architected as follows:

- 40 x 17TB FlashBlade with redundant external fabric odules (XFM)
- External fabric modules connected to Arista Leaf Switches via eight 100GbE configured as a single LACP/MLAG group

System Configuration

This section describes the system configuration tested by Graphcore and Pure Storage for the IPU-POD64 reference platform.

First, we'll cover the configuration settings for Poplar host, storage, and networking.

Poplar Host Configuration

The Poplar framework can be used to define graph operations and control the execution and profiling of code on the IPU, as well as to configure the host. For this configuration, a single 100G Ethernet interface for the storage connection configured with a Mellanox OFED device driver.

An example fstab entry is as follows:

```
10.12.70.250:/public1 /mnt/public nfs
ro,relatime,vers=3,hard,proto=tcp,nconnect=16,timeo=600,retrans=2,sec=sys,mountvers=3,mountport=2049
,mountproto=udp,rsize=524288,wsize=524288,namlen=255 0 0
```

NOTE: nconnect=16 is required to achieve the stated performance and is only available on the 5.x kernel.

Storage Configuration

To configure the storage system, the following tasks need to be performed:

1. Create the file system that will host the data.
2. Add the NFS protocol to the file system (NFSv3 was used, although NFSv4.1 also available).
3. Configure the file system so that the export options are applied to define which clients can mount the file system.

This can be performed via the GUI.

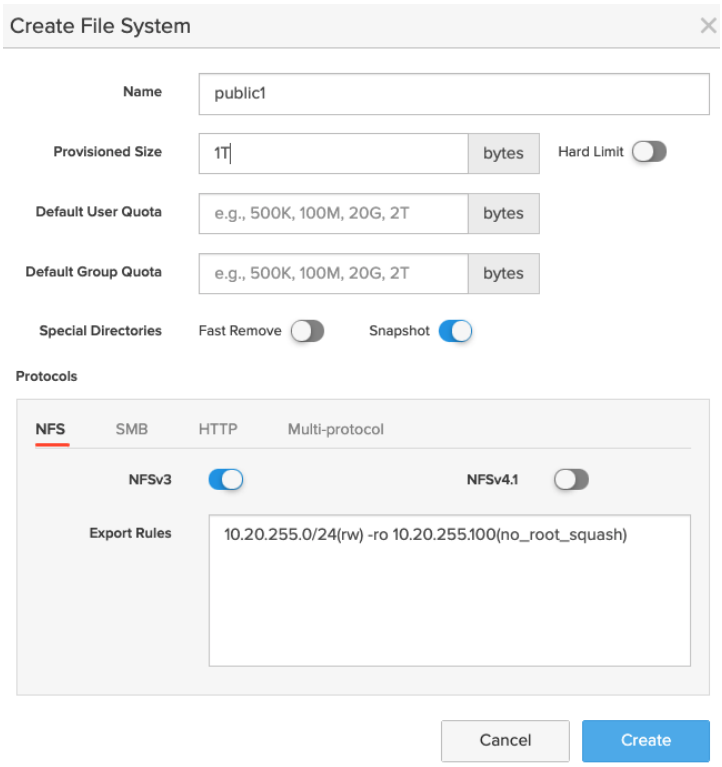

Figure 6. Creating the file system and adding the NFS protocol.

The above steps can also be carried out via the command line interface (CLI) when logged into the FlashBlade Management interface as pureuser:

```
purefs create --size 1t public1
purefs add --protocol nfsv3 public1
purefs nfs setattr --rules '10.20.255.0/24(rw) -ro 10.20.255.100(no_root_squash)' public1'
```

There is no other performance tuning, data protection, or advanced setting required to optimize the performance of this file system across the 40 x 17TB blades. This is managed automatically within Pure FlashBlade.

Networking Configuration

The following clients are connected to the Pure FlashBlade system via a VLAN network:

- MTU 1500 on 100G VLAN
- IPv4 subnets defined in the Pure management interface with appropriate VLAN
- 100G VLAN configured in ToR switch to provide untagged connection to client
- Two four-way MLAG using Arista MLAG active/active technology from ToR to pair of SPINE switches
- Two four-way MLAG using Arista MLAG active/active technology from storage LEAF to pair of SPINE switches

Test and Performance Results

We evaluated the Graphcore and Pure Storage infrastructure solution for its suitability for various deep learning workloads. The testing highlighted the high read IOPS and the throughput achieved with performance tests on ResNet 50 and BERT. Further testing revealed near-linear scaling as we increased the number of jobs run on the infrastructure. This section highlights some of the results from tests performed in our labs.

Performance tests were undertaken using:

- ResNet 50
- BERT-Large
- Standardized storage benchmark for AI workloads
- Fio (Flexible I/O tester)

For the ResNet and BERT tests, it is important to keep in mind that the focus is on IPU performance. The FlashBlade system needs to be fast enough to keep the processors fed with all the data they can consume. FlashBlade can deliver up to 1GiB/s per blade, so the peak throughput measurements do not approach the limits of the 40-blade system.

ResNet 50

ResNet 50 tests were performed using eight hosts and produced the following results:

	Read (MiB/s)	Read IOPS	Total Runtime (s)	Throughput (items/sec)
Mean	7990.29	47943.16	247.90	16812.19
Max	15652.60	93933.00	258.55	16851.15

Table 1. ResNet 50 performance tests.

BERT=Large

Below are the results of testing BERT pretrain large using four hosts.

	Read (MiB/s)	Read IOPS	Total Runtime (s)	Throughput (items/sec)
Mean	642.9614	1286.038	2238.935	2404.961
Max	1880.5	3761	2378.873	3759.419

Table 2. BERT-Large performance tests.

Standardized Storage Benchmark for AI Workload

Benchmark results for the AI Image workload are presented in the figure and table below.

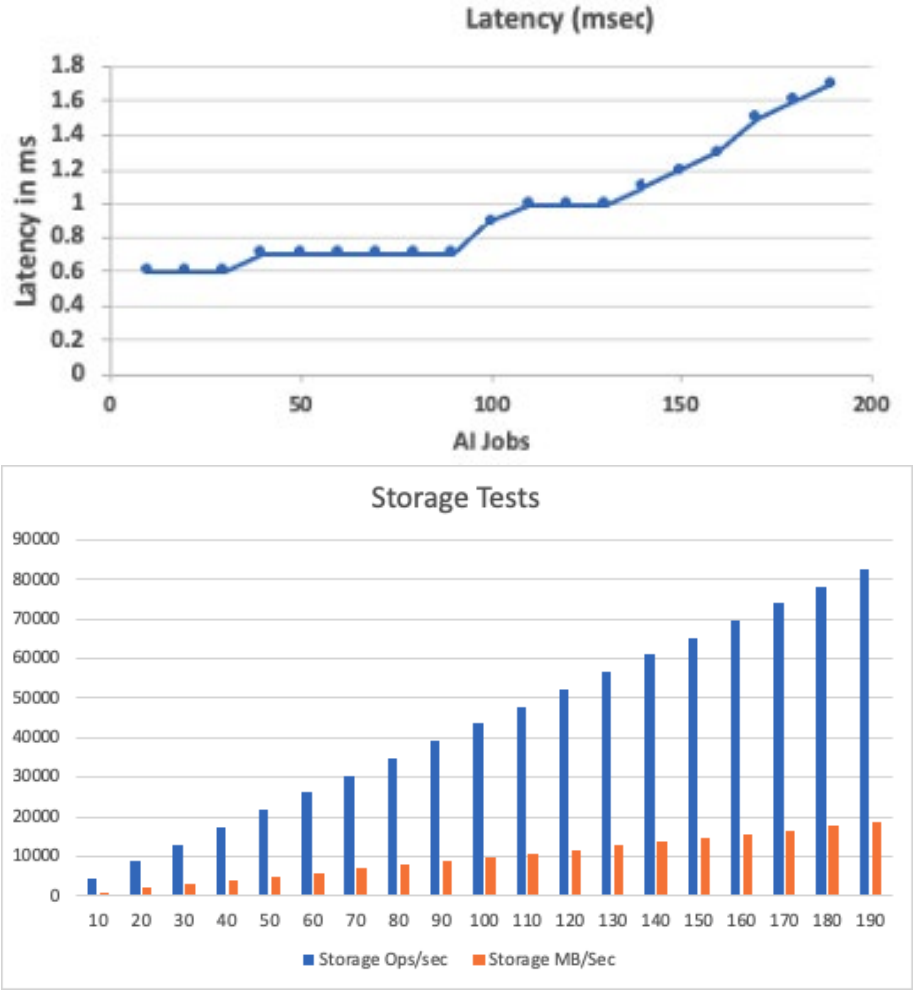

Figure 7. Benchmark results for the AI Image workload.

AI Jobs	Latency (msec)	Storage Ops/sec	Storage MB/sec
10	0.6	4350	978
20	0.6	8701	1956
30	0.6	13051	2934
40	0.7	17402	3912
50	0.7	21752	4891
60	0.7	26103	5869
70	0.7	30453	6845
80	0.7	34804	7827
90	0.7	39155	8803
100	0.9	43505	9780
110	1.0	47856	10758
120	1.0	52206	11737
130	1.0	56557	12717
140	1.1	60907	13692
150	1.2	65258	14671
160	1.3	69609	15647
170	1.5	73959	16625
180	1.6	78309	17600
190	1.7	82660	18581

Table 3. Benchmark results for the AI Image workload.

Fio and Application-level Sanity Checks

In addition to the SPEC testing, we performed many manual tests using Fio, Python client applications, and object-storage API protocols.

For NFS, six Poplar hosts read with an aggregate bandwidth of 37GB/s against the expected max of 40GB/s that the 40 x 17TB FlashBlade system was theoretically capable of sustaining. A single Poplar host can reliably read 10GB/s with Fio using TCP (no RDMA, no jumbo frames) without any other contention. For S3 object storage, six Poplar hosts read an aggregate bandwidth of ~24GB/s.

Conclusion

Artificial intelligence is a data-centric field whose storage requirements are both highly demanding and somewhat idiosyncratic relative to compute, as we have known previously. Delivering best-in-class performance for AI requires highly optimized solutions co-designed by compute system and storage providers.

Graphcore and Pure Storage have worked closely together to develop this reference architecture, drawing on deep expertise in their respective fields. The coming together of the intelligence processing unit (IPU) and FlashBlade, with its wide range of file system protocols, gives users flexibility and choice. They have the assurance that, no matter howthey configure their setup, it will deliver superlative AI performance.

Both companies' technologies—and artificial intelligence in general—are advancing quickly, and the ongoing partnership between Graphcore and Pure Storage will ensure that every advance and technological refinement we make in future will be in close collaboration, for the benefit of our mutual customers and their evolving AI needs

Pure Storage, Inc.
650 Castro Street, #400
Mountain View, CA 94041