

STORAGE DEVELOPER CONFERENCE



Fremont, CA
September 12-15, 2022

BY Developers FOR Developers

A **SNIA** Event

Accelerating Near Real-Time Analytics with High Performance Object Storage

Mayank Saxena

Senior Director of Engineering

Data Fabric Solutions,
Memory Solutions Lab, Samsung

www.samsungmsl.com

Sept 14th 2022

HPOS with Computational SSD Acceleration

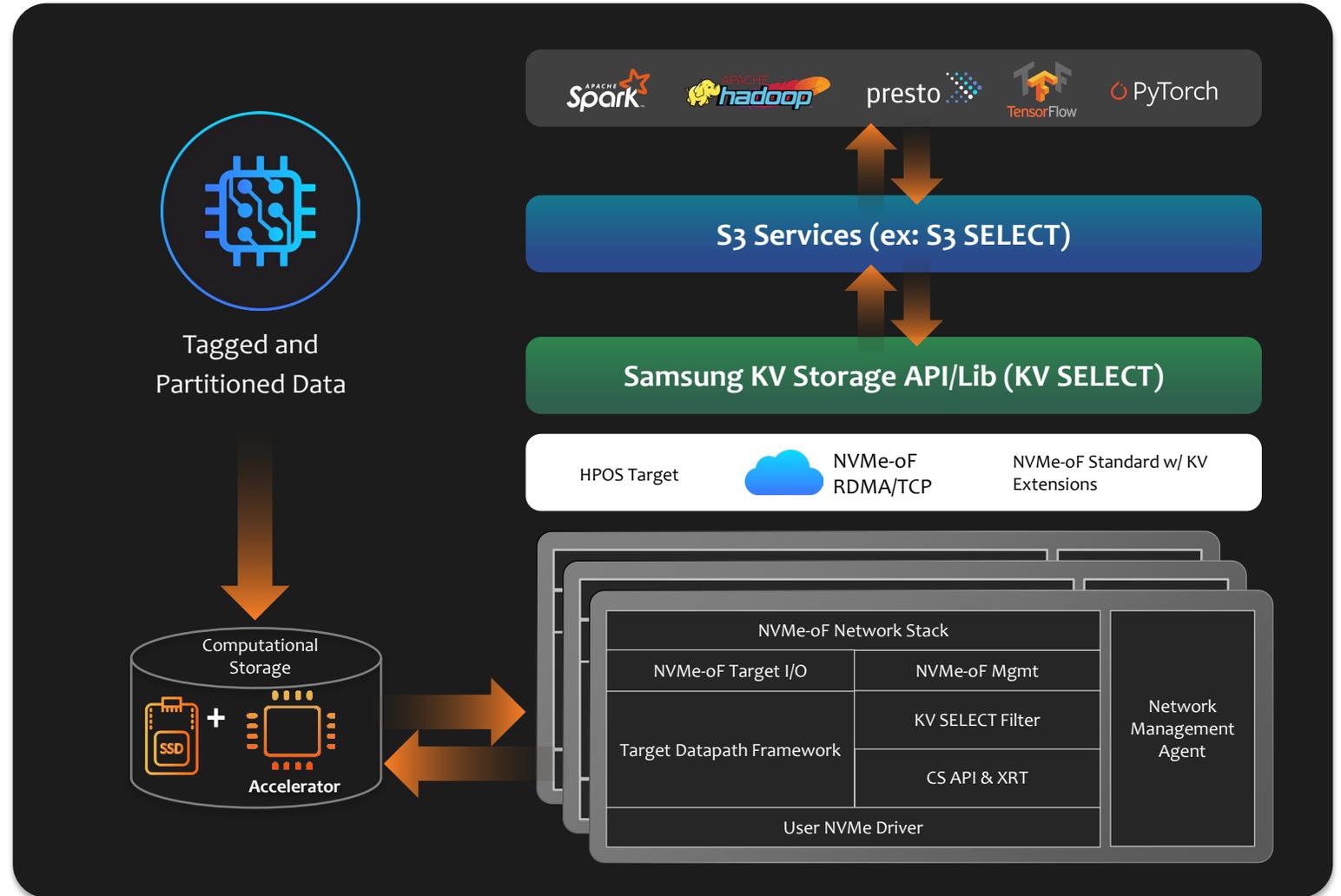
High Performance Object Storage (HPOS)

Use Cases

Large scale real time analytics - Smart City, Smart Home, eHealth, IoT, Images, Video Analytics, Security

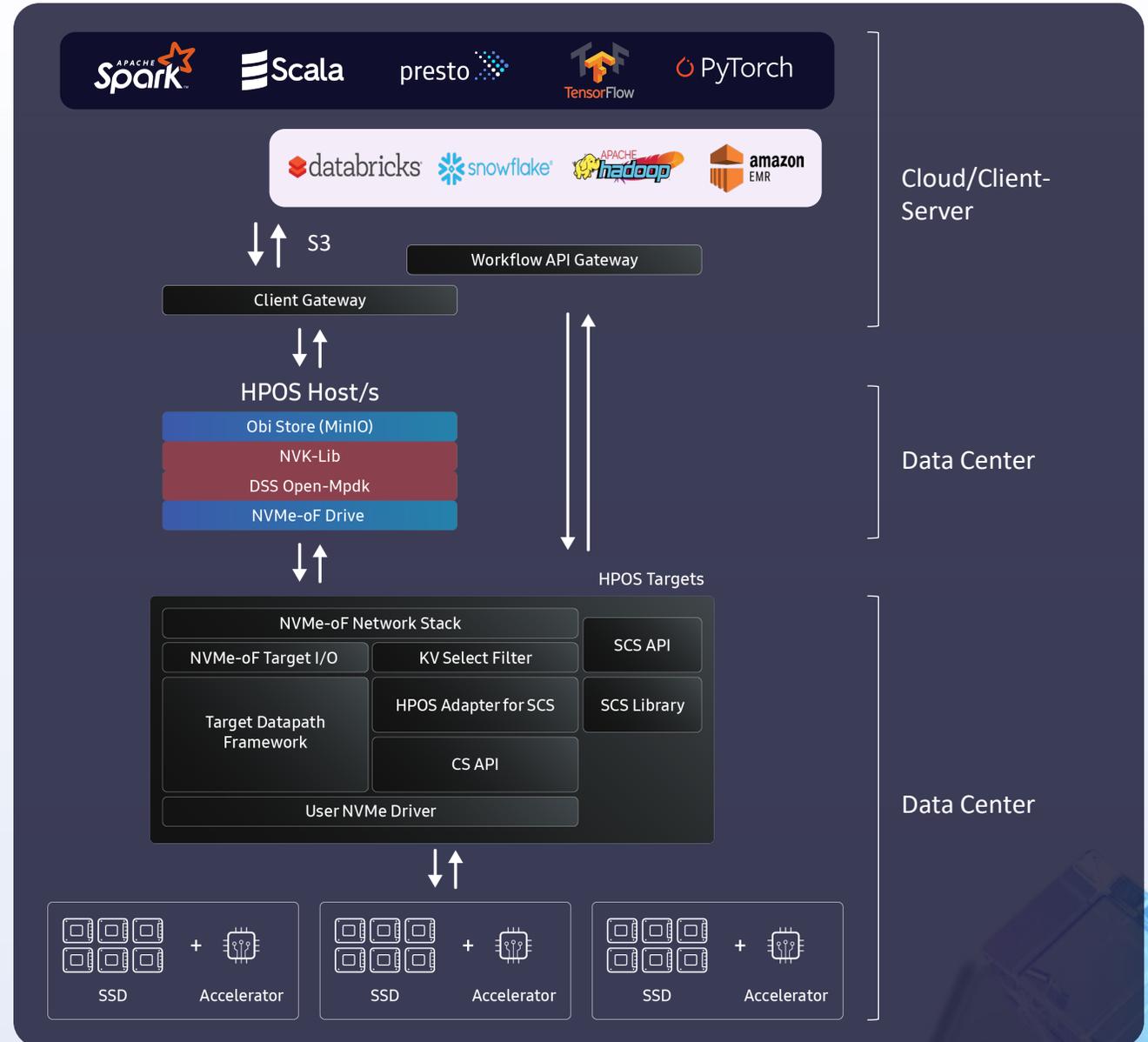
Benefits

- **Faster** queries
- **Lesser** network traffic
- **Lower TCO** due to reduced CPU and network traffic

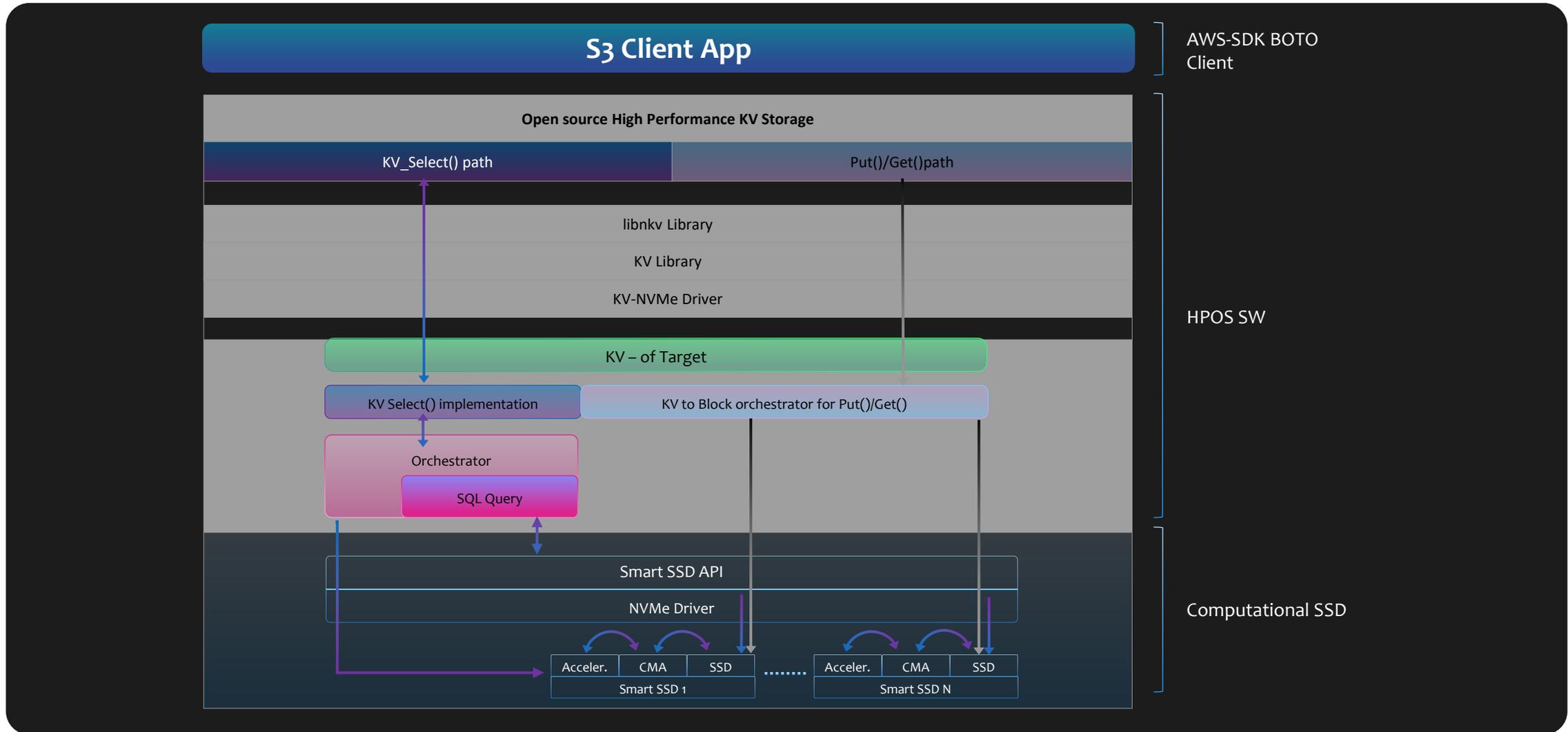


Efficient & Accelerated Analytics Solutions

- Custom Functions in AI pipeline accelerated by HPOS + SmartSSD
- Client applications like (e.g. S3 Select, AI apps) talk to API Gateway for I/O and HPOS workflow definition
- Deployment efficiency - HPOS Host & Target as Containers
- Scale - Client Gateway and API gateway can be in the Cloud or Datacenter
- HPOS CSD based Solutions
 - S3 Select – 100x more Throughput
 - Video AI – 51x faster Inferencing
 - Significant Lesser TCO



End-to-End S3 Select with HPOS

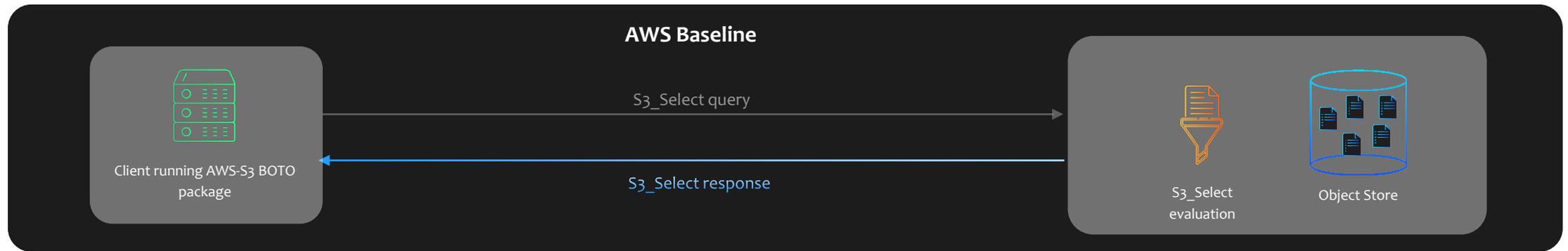


→ Normal data path

→ Path for SmartSSD offload

CMA Computational Memory Area

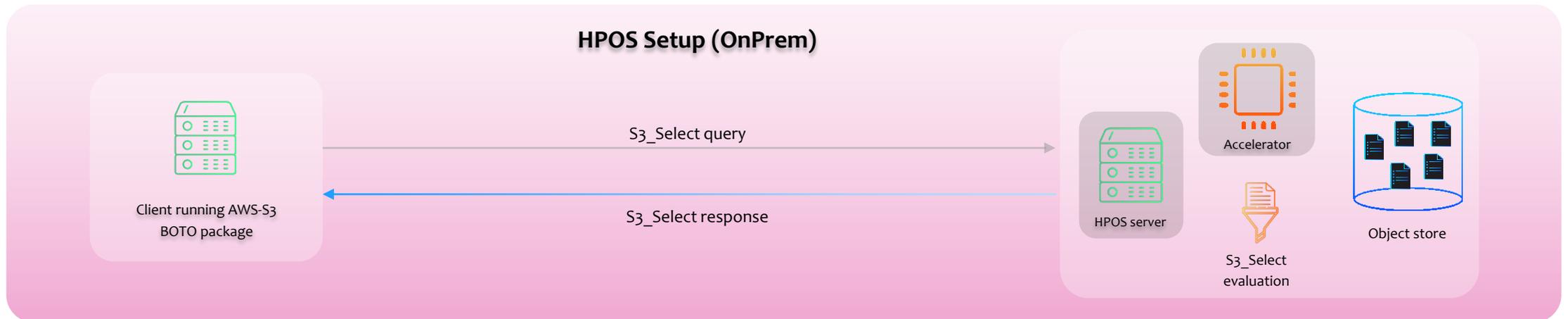
Near data processing for S3_Select



Customer's Sample Dataset

Index	Year	DayOfYear	Precipitation	ETO	tMax	tMin	RS
0	1	1	0	0	-26.6808	-29.7293	0
1	1	2	0	0	-27.0202	-30.9575	0
2	1	3	0	0	-28.0491	-31.0459	0
3	1	4	0	0	-26.2893	-29.3781	0
4	1	5	0	0	-25.1941	-27.9265	0
5	1	6	0	0	-23.5173	-27.2431	0
6	1	7	0	0	-23.8696	-29.4691	0
7	1	8	0	0	-24.4267	-29.0663	0

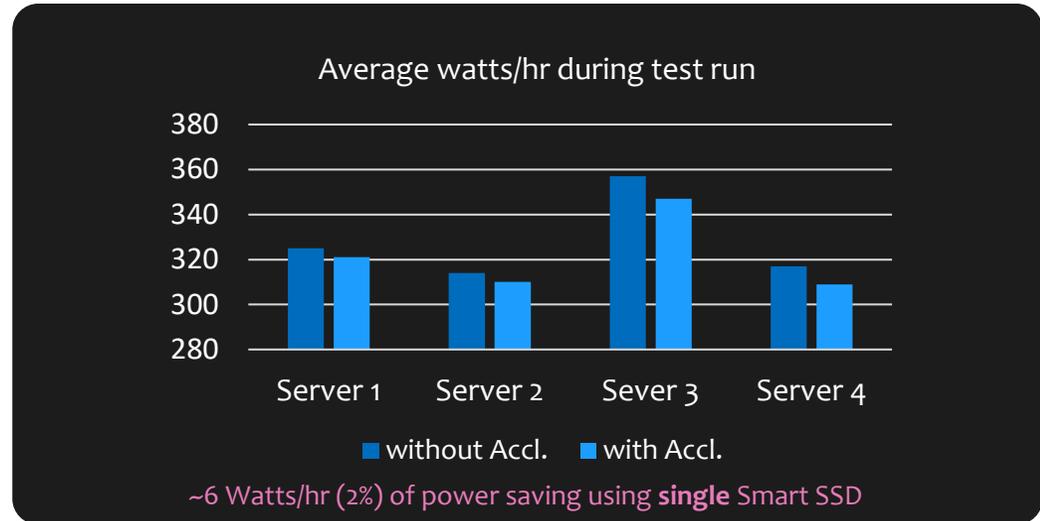
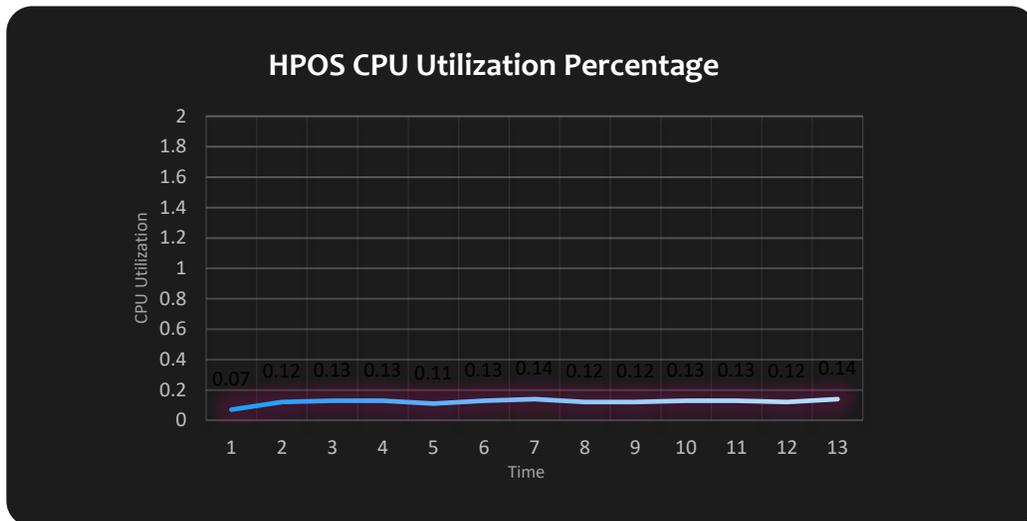
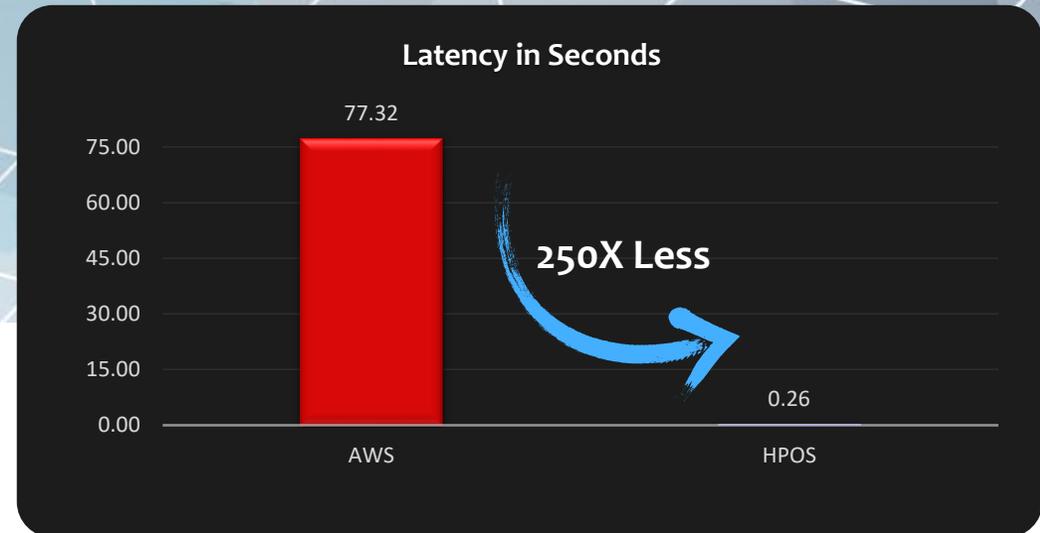
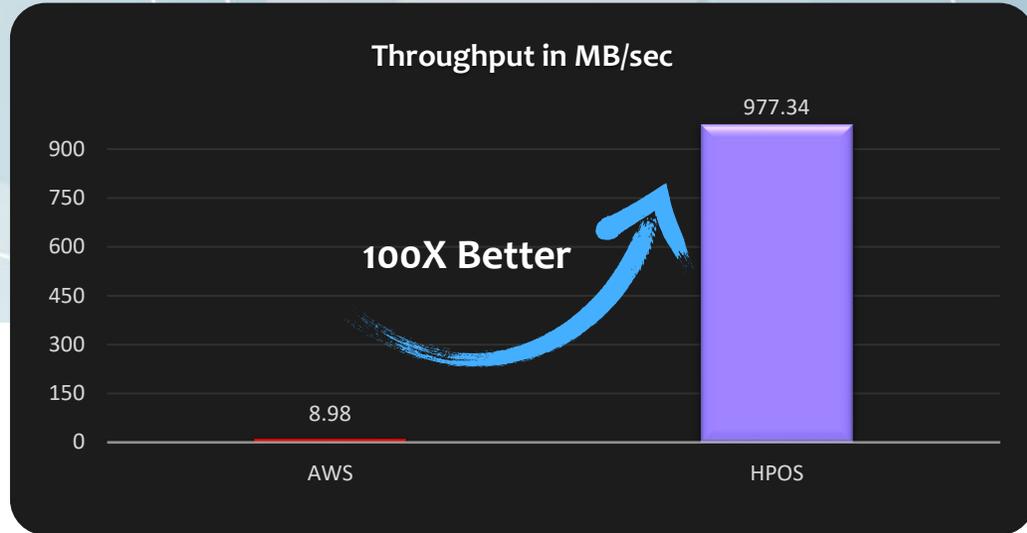
S3_Select Query ('Select COUNT(*) FROM S3Object s WHERE dayofyear < 55');



HPOS S3_Select Drag Race Demo

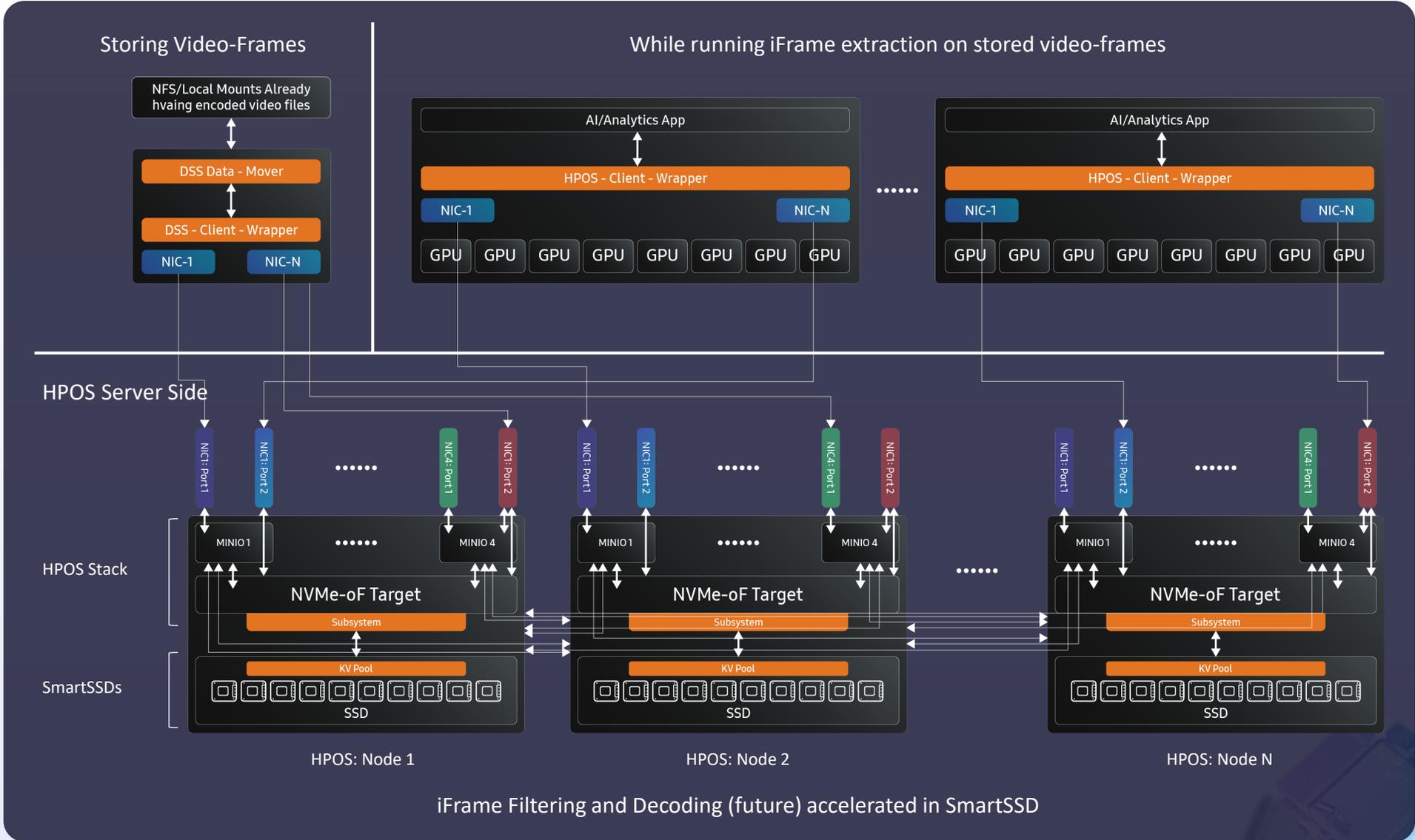


AWS S3 Select vs HPOS S3 Select



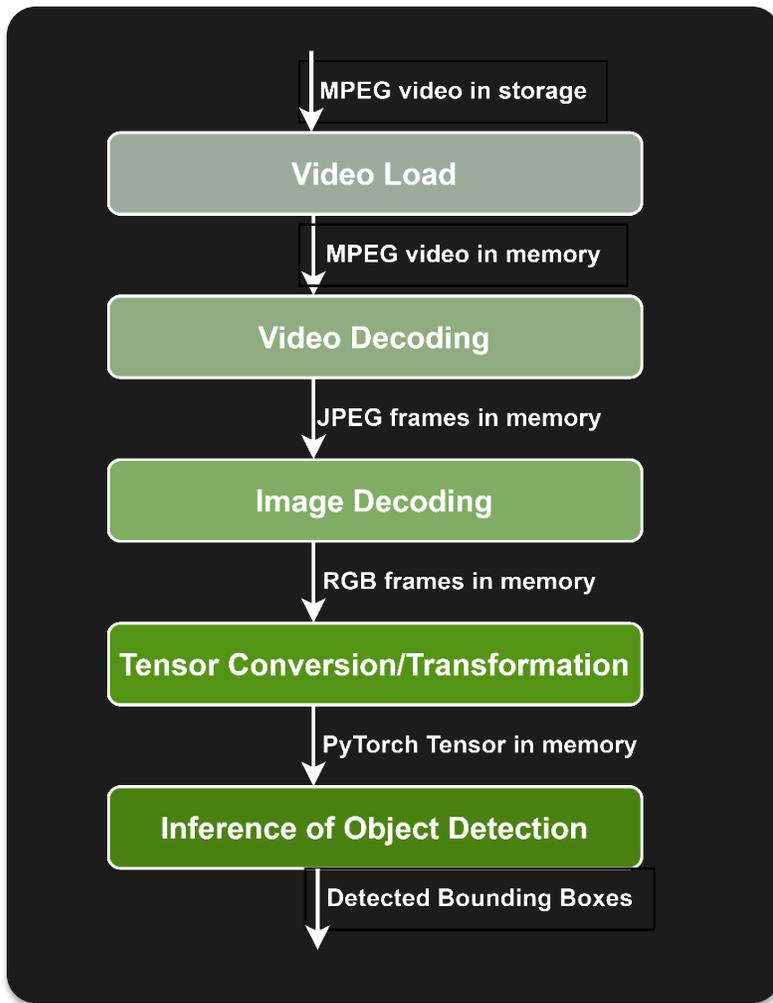
High Performance Video Analytics – Solution Deployment

- High Performance Video Analytics
- TCO – Disaggregation of GPUs and Storage
- Smart SSD acceleration for video decoding (future) and iFrame extraction
- Large Object Support (Video files)
- Supports SmartSSD 1.0 & Smart SSD 2.0



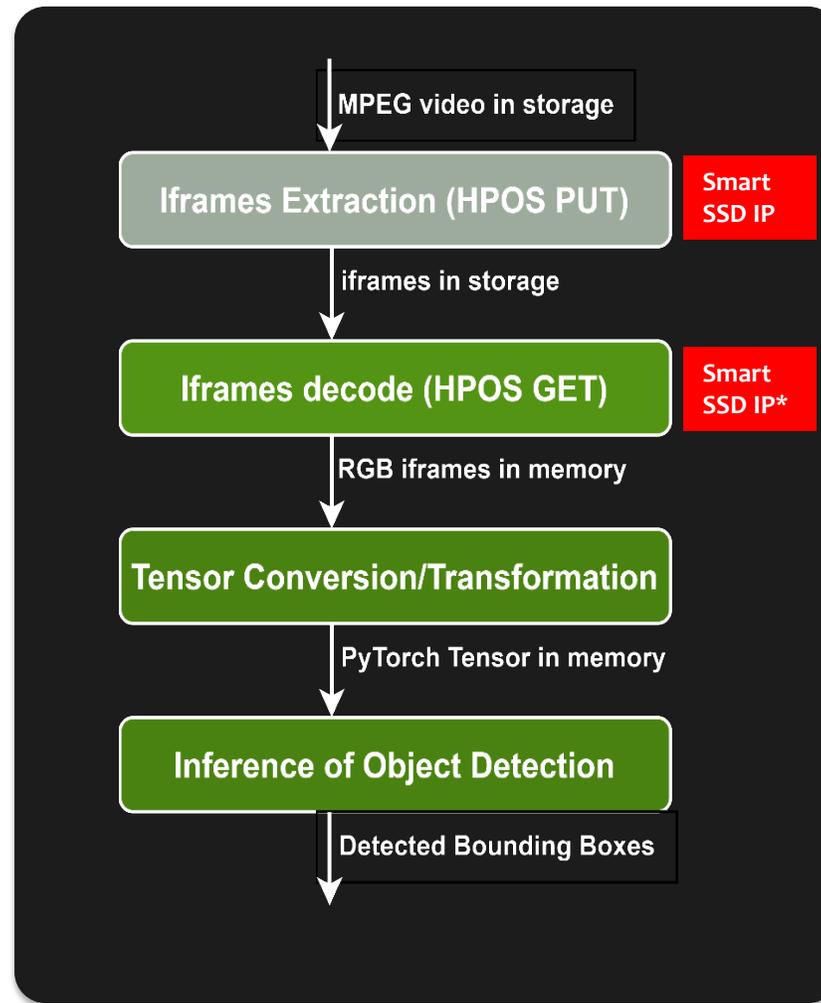
Video Analytics Pipeline Comparison

Generic* pipeline



VS

HPVA# pipeline



HPOS Target

AI/ML application
(client machine)

* Used NFS for storage
HPOS with SmartSSD as storage

Drag Race – Generic vs HPVA

Generic Pipeline

The screenshot shows two terminal windows. The left window displays the execution of a script for the Generic Pipeline, including file operations and the execution of inference and preprocessing tasks. The right window shows the output of the `top` command, which provides a real-time view of system resource usage.

Overall inference time: 61.18115997314453 (s)

Preprocessing time: 2.4198296070898877 (s)

Preprocessing CPU %: 36.74

Time	CPU	user	nice	system	siowait	steal	idle
04:49:43 PM	all	33.05	0.00	0.03	0.00	0.00	66.92
04:49:44 PM	all	38.98	0.00	0.02	0.00	0.00	61.00
04:49:45 PM	all	39.35	0.00	0.06	0.00	0.00	60.60
04:49:46 PM	all	39.25	0.00	0.09	0.00	0.00	60.65
04:49:47 PM	all	39.80	0.00	0.14	0.00	0.00	60.07
04:49:48 PM	all	39.84	0.00	0.09	0.00	0.00	60.87
04:49:49 PM	all	39.22	0.00	0.10	0.00	0.00	60.68
04:49:50 PM	all	40.25	0.00	0.06	0.00	0.00	59.69
04:49:51 PM	all	39.83	0.00	0.06	0.00	0.00	60.11
04:49:52 PM	all	39.01	0.00	0.02	0.00	0.00	60.06
04:49:53 PM	all	39.61	0.00	0.15	0.00	0.00	60.25
04:49:54 PM	all	39.77	0.00	0.09	0.00	0.00	59.34
04:49:55 PM	all	40.08	0.00	0.06	0.00	0.00	59.86
04:49:56 PM	all	38.77	0.00	0.02	0.00	0.00	61.21
04:49:57 PM	all	37.78	0.00	0.13	0.00	0.00	67.10
04:49:58 PM	all	4.53	0.00	0.18	0.00	0.00	95.29
04:49:59 PM	all	0.00	0.00	0.00	0.00	0.00	100.00
04:50:00 PM	all	0.00	0.00	0.01	0.00	0.00	99.99
04:50:01 PM	all	0.00	0.00	0.01	0.00	0.00	99.99
04:50:02 PM	all	0.02	0.00	0.06	0.00	0.00	99.92
04:50:03 PM	all	0.00	0.00	0.01	0.00	0.00	99.99
04:50:04 PM	all	0.00	0.00	0.00	0.00	0.00	100.00
04:50:05 PM	all	0.00	0.00	0.00	0.00	0.00	100.00
04:50:06 PM	all	0.00	0.00	0.00	0.00	0.00	100.00
04:50:07 PM	all	0.01	0.00	0.00	0.00	0.00	99.99
04:50:08 PM	all	0.00	0.00	0.01	0.00	0.00	99.99
04:50:09 PM	all	0.00	0.00	0.00	0.00	0.00	100.00
04:50:09 PM	CPU	user	nice	system	siowait	steal	idle
04:50:10 PM	all	0.00	0.00	0.01	0.00	0.00	99.99
04:50:11 PM	all	0.00	0.00	0.00	0.00	0.00	100.00
04:50:12 PM	all	1.25	0.00	5.82	0.00	0.00	92.93
04:50:13 PM	all	2.62	0.00	0.24	0.00	0.00	97.14
04:50:14 PM	all	4.88	0.00	5.05	0.00	0.00	90.16
04:50:15 PM	all	34.71	0.00	0.00	0.00	0.00	64.94
04:50:16 PM	all	36.74	0.00	0.00	0.00	0.00	63.05
04:50:17 PM	all	15.96	0.00	0.00	0.00	0.00	83.86
04:50:18 PM	all	0.01	0.00	0.01	0.00	0.00	99.98
04:50:19 PM	all	0.00	0.00	0.00	0.00	0.00	100.00

HPVA Pipeline

The screenshot shows two terminal windows. The left window displays the execution of a script for the HPVA Pipeline, including file operations and the execution of inference and preprocessing tasks. The right window shows the output of the `top` command, which provides a real-time view of system resource usage.

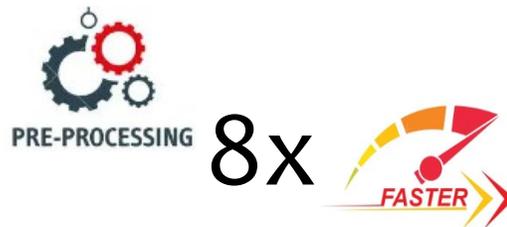
Overall inference time: 1.2296864986419678 (s)

Preprocessing time: 0.286761045459326 (s)

Preprocessing CPU %: 4.24

Time	CPU	user	nice	system	siowait	steal	idle
04:58:23 PM	CPU	user	nice	system	siowait	steal	idle
04:58:24 PM	all	0.00	0.00	0.01	0.00	0.00	99.99
04:58:25 PM	all	0.00	0.00	0.00	0.00	0.00	100.00
04:58:26 PM	all	0.00	0.00	0.01	0.00	0.00	99.99
04:58:27 PM	all	0.00	0.00	0.00	0.00	0.00	100.00
04:58:28 PM	all	0.01	0.00	0.01	0.00	0.00	99.98
04:58:29 PM	all	0.38	0.00	0.06	0.00	0.00	99.57
04:58:30 PM	all	1.33	0.00	5.96	0.00	0.00	92.71
04:58:31 PM	all	6.21	0.00	0.34	0.00	0.00	93.45
04:58:32 PM	all	0.73	0.00	3.25	0.00	0.00	96.02
04:58:33 PM	all	0.06	0.00	0.10	0.00	0.00	99.84
04:58:34 PM	all	0.00	0.00	0.00	0.00	0.00	100.00
04:58:35 PM	all	0.02	0.00	0.03	0.00	0.00	99.94
04:58:36 PM	all	0.08	0.00	0.02	0.00	0.00	99.88
04:58:37 PM	all	5.42	0.00	1.23	0.00	0.00	93.35
04:58:38 PM	all	33.36	0.00	0.81	0.01	0.00	65.82
04:58:39 PM	all	0.19	0.00	0.11	0.00	0.00	99.69
04:58:40 PM	all	0.00	0.00	0.00	0.00	0.00	100.00
04:58:41 PM	all	0.02	0.00	0.06	0.00	0.00	99.92
04:58:42 PM	all	0.05	0.00	0.07	0.00	0.00	99.89
04:58:43 PM	all	0.00	0.00	0.00	0.00	0.00	100.00
04:58:44 PM	all	0.02	0.00	0.05	0.00	0.00	99.93
04:58:45 PM	all	0.00	0.00	0.00	0.00	0.00	100.00
04:58:46 PM	all	0.00	0.00	0.02	0.00	0.00	99.98
04:58:47 PM	all	0.02	0.00	0.05	0.00	0.00	99.93
04:58:48 PM	all	0.01	0.00	0.05	0.00	0.00	99.94
04:58:49 PM	all	0.00	0.00	0.00	0.00	0.00	100.00
04:58:50 PM	all	0.01	0.00	0.00	0.00	0.00	99.99
04:58:51 PM	all	4.24	0.00	5.77	0.00	0.00	92.86
04:58:52 PM	all	1.38	0.00	0.19	0.00	0.00	95.56
04:58:53 PM	all	4.24	0.00	3.49	0.00	0.00	93.69
04:58:54 PM	all	2.82	0.00	0.00	0.00	0.00	100.00
04:58:55 PM	all	0.00	0.00	0.00	0.00	0.00	100.00
04:58:56 PM	all	0.00	0.00	0.00	0.00	0.00	100.00
04:58:57 PM	all	0.00	0.00	0.00	0.00	0.00	100.00
04:58:58 PM	all	0.00	0.00	0.01	0.00	0.00	99.99
04:58:59 PM	all	0.56	0.00	0.08	0.00	0.00	99.36
04:59:00 PM	all	0.35	0.00	0.19	0.00	0.00	99.45

Profiling Comparison Summary



	Generic (Baseline)	HPVA	Comment
Video IOs	video file 936 KB (SSD to Target to Host)	11 iframes, ~20KB/frame (SSD drive to Decoder in Target) 11 decoded frames, ~220KB/decoded Frames (Target to Host)	More performance gain for video files with lesser number of iFrames
Avg CPU Utilization (preprocessing)	~29%	~2.8%	Key TCO benefit with HPOS, as preprocessing done in target (with SmartSSD) itself.
Avg CPU Utilization (overall inference)	~38%	33%	Currently after preprocessing it's the same pipeline. So, not much difference.*
Latency (preprocessing)	2.42s	0.29s	Latency benefit with HPOS, as preprocessing done in target (with SmartSSD) itself
Latency (overall inference)	61.18s	1.2s	After preprocessing it's the same pipeline, but less frames (only iFrames) to process for HPOS.

Open Source – HPOS Foundation

Open Source <https://github.com/OpenMPDK/DSS>

- DSS – Disaggregate Storage Solution

Complete Ecosystem

- AI Benchmarking Framework supporting user preferred training and models
- Client Wrappers supporting Pytorch and Tensorflow
- Host and Target Stack

More CSD specific loadable modules & SDK coming up ...



Thank You

Mayank Saxena

Senior Director of Engineering

mayank.s4@samsung.com

Data Fabric Solutions,

Memory Solutions Lab, Samsung

www.samsungmsl.com