NVMe Computational Storage

Standardizing offload of computation via NVMe

Presented by Kim Malone, Intel and Stephen Bates, Eideticom

Agenda

- NVMe Computational Storage Architecture
- Example Flows
- NVMe changes for computational storage
- NVMe Computational Storage Task Group

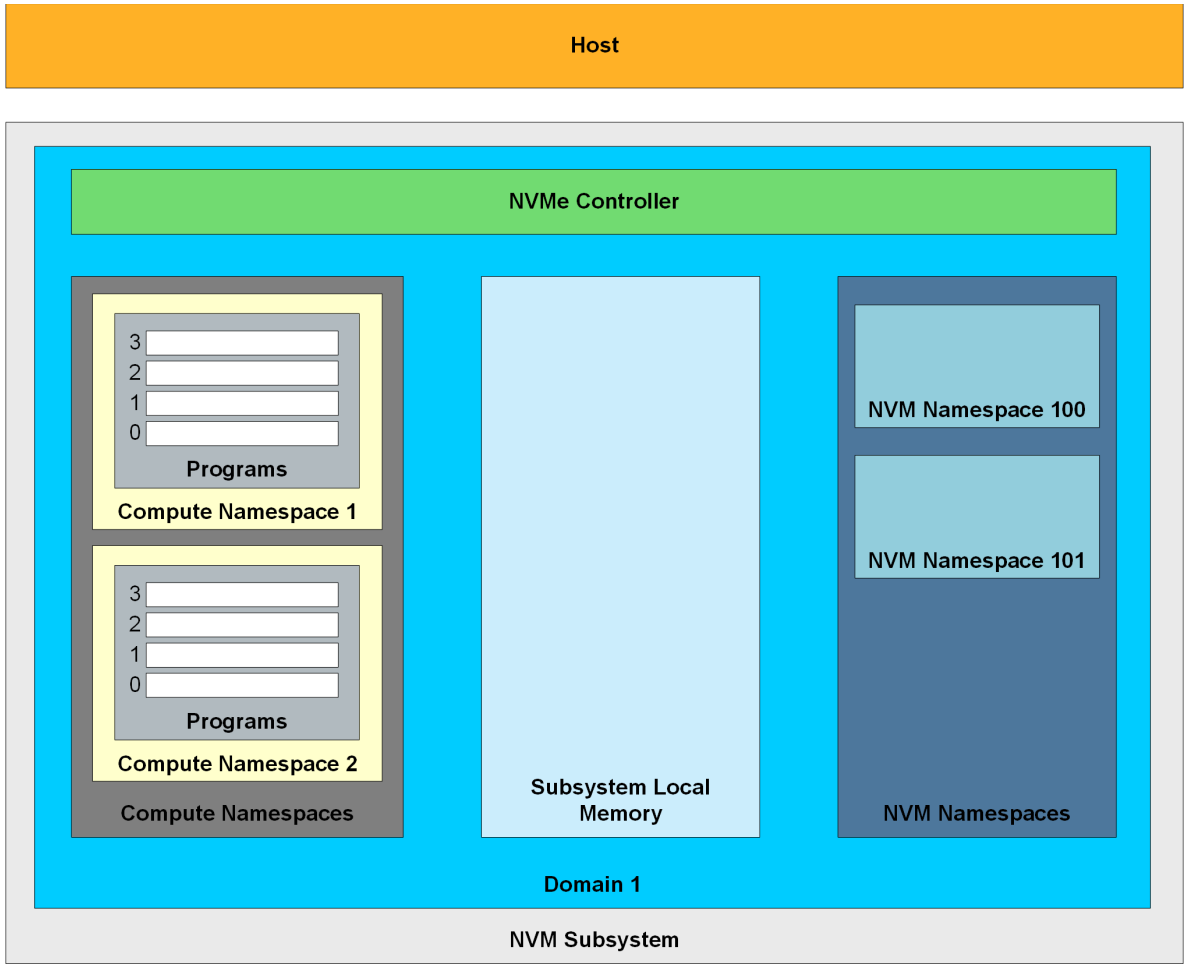
 ©2022 Storage Networking Industry Association ©. Intel, Eideticom. All Rights Reserved.

NVMe Computational Storage Architecture

An extensible architecture

 ©2022 Storage Networking Industry Association ©. Intel, Eideticom. All Rights Reserved.

Major Architectural Components



- Programs operate only on data in Subsystem Local Memory
  - Includes program input, output
  - Data is copied between Subsystem Local Memory and host memory using new NVMe commands
  - The details of commands to copy data between Subsystem Local Memory and NVM namespaces are TBD

This presentation discusses NVMe work in progress, which is subject to change without notice.

 ©2022 Storage Networking Industry Association ©. Intel, Eideticom. All Rights Reserved.

Programs as Computational Storage Offloads

Programs:

- Invoked and used in a standard way
  - Conceptually similar to software functions
  - Called with parameters and run to completion
- Operate on data in on-device memory
- Run on compute resources
- May be in hardware or software
  - Device may offer fixed function programs, or
  - Downloadable in hardware agnostic bytecode (eBPF) from host for later execution
- A program may only be able to execute on a subset of the compute resources in an NVM subsystem

This presentation discusses NVMe work in progress, which is subject to change without notice.

 ©2022 Storage Networking Industry Association ©. Intel, Eideticom. All Rights Reserved.

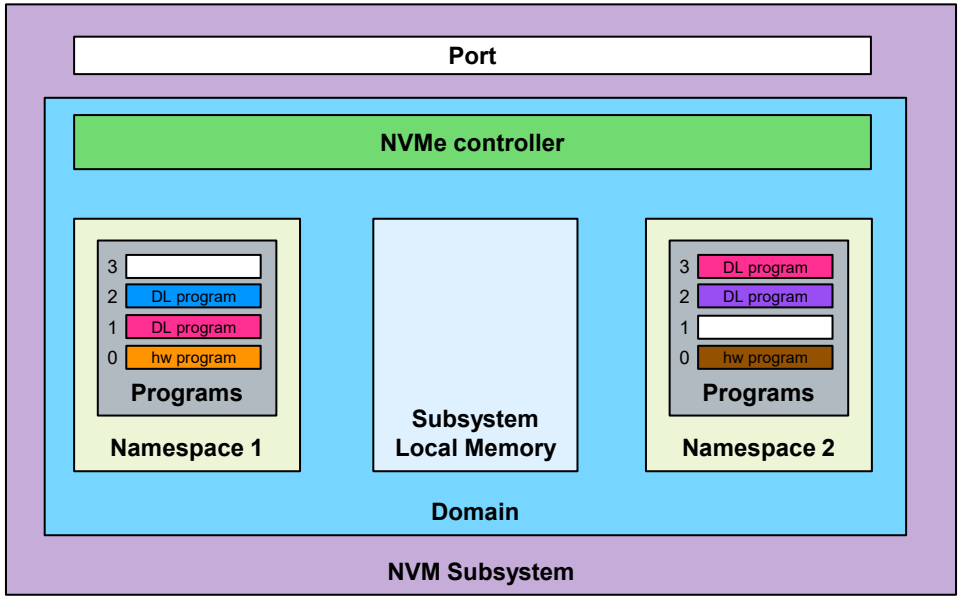
Namespaces for Computational Programs Command set

This command set introduces a new namespace type: a computational namespace.

A computational namespace:

- Is an entity in an NVMe subsystem that is able to execute one or more programs
- May have asymmetric access to subsystem memory
- May support a subset of all possible program types

Programs are managed per computational namespace



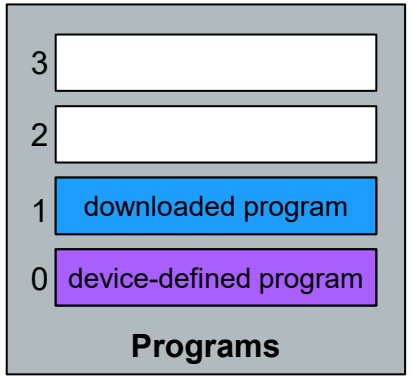
(Drawing contains an example configuration, not intended to convey all cases)

This presentation discusses NVMe work in progress, which is subject to change without notice.

 ©2022 Storage Networking Industry Association ©. Intel, Eideticom. All Rights Reserved.

Downloadable and device-defined programs

- Support for both device-defined and downloadable programs
- Device-defined programs
  - “Fixed” programs provided by the manufacturer
  - Functionality implemented by the device that are callable as programs
  - e.g. compression, decryption
- Downloadable programs
  - Programs that are loaded to the Computational Programs namespace by the host

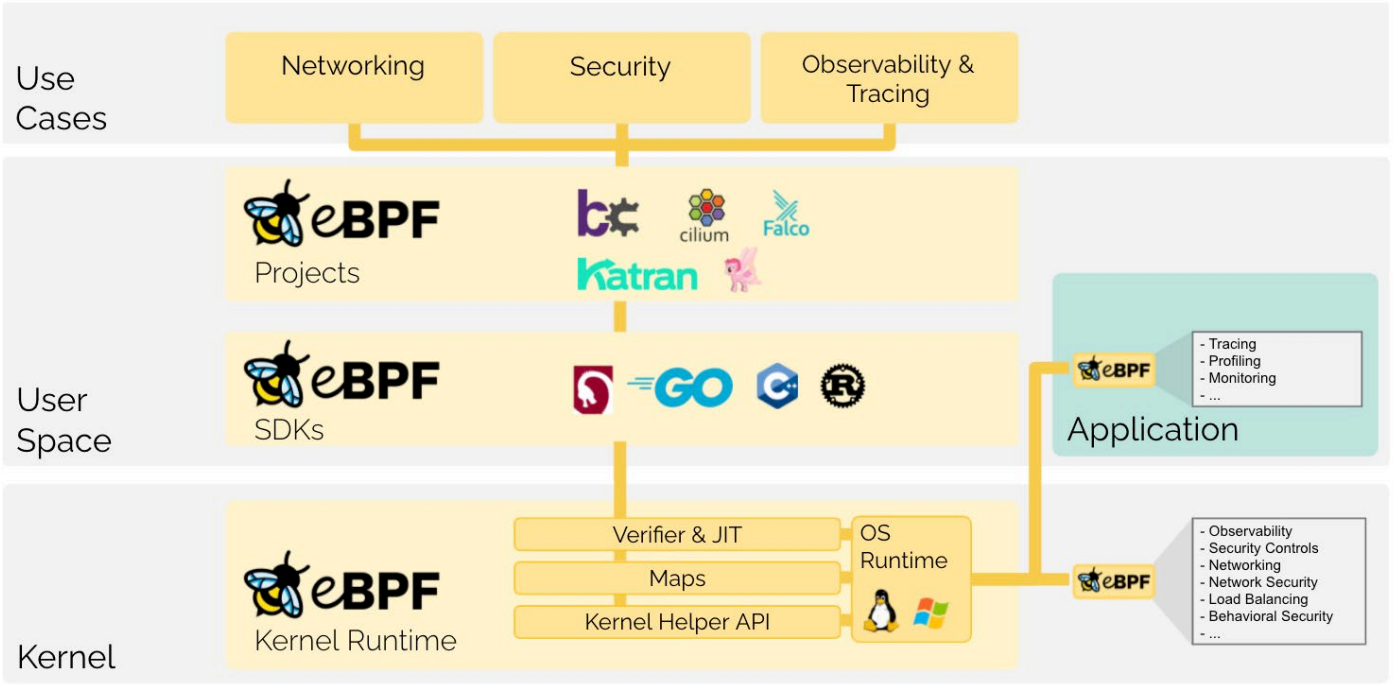


This presentation discusses NVMe work in progress, which is subject to change without notice.

 ©2022 Storage Networking Industry Association ©. Intel, Eideticom. All Rights Reserved.

eBPF for Downloadable Programs

- Why downloadable programs?
  - Flexibility
  - Process complex formats
  - Emerging applications
  - Portability from existing applications
- Why eBPF?
  - Vendor Agnostic
  - Well understood
  - Existing ecosystems
  - LLVM
  - Toolchains
  - Sits under Linux Foundation



This presentation discusses NVMe work in progress, which is subject to change without notice.

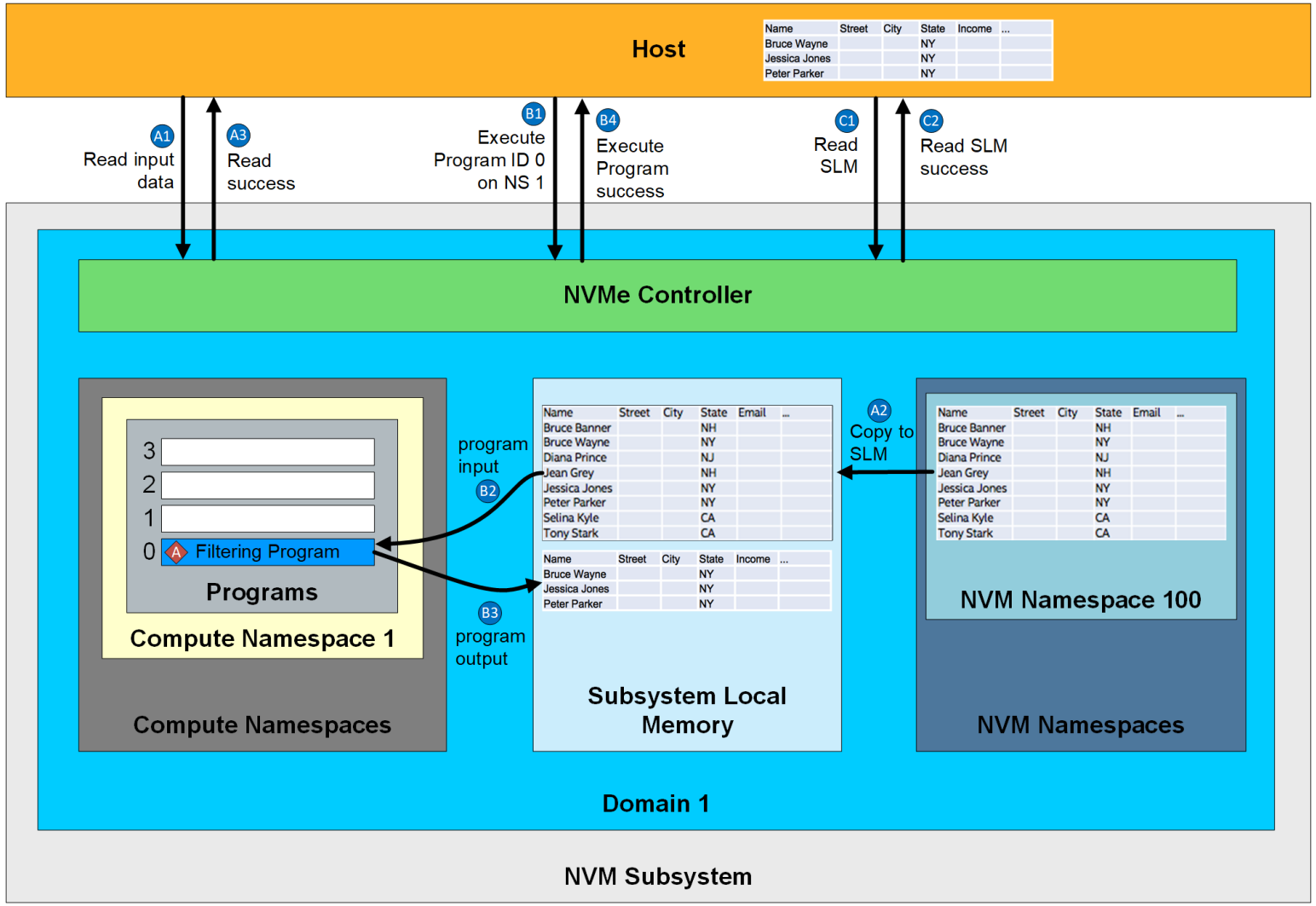
 ©2022 Storage Networking Industry Association ©. Intel, Eideticom. All Rights Reserved.

Example Flows

How does it work?

 ©2022 Storage Networking Industry Association ©. Intel, Eideticom. All Rights Reserved.

Flow: Execute Program – Simple Data Filter



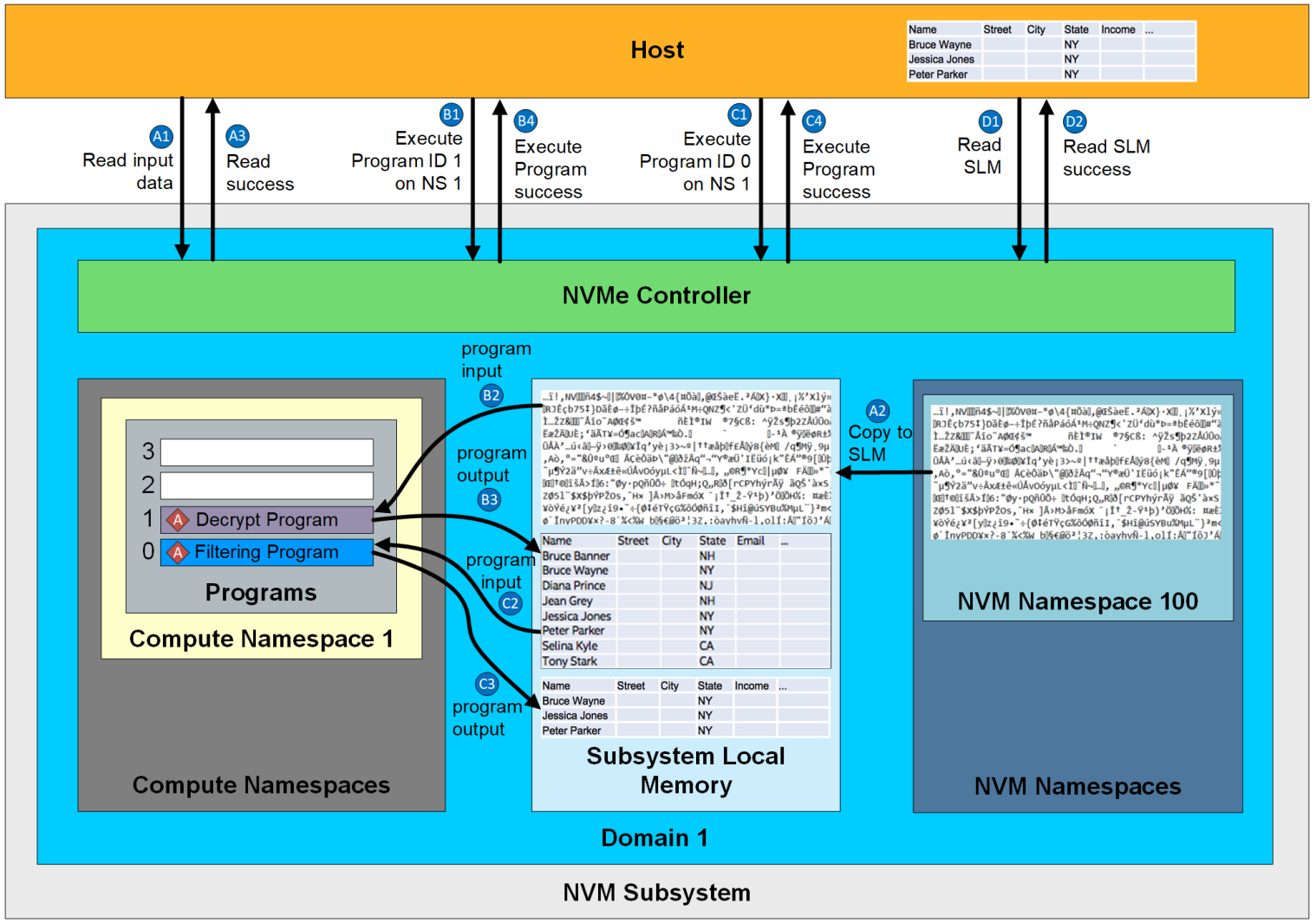
Flow steps

- A "Read" stored data into subsystem memory
- B Execute Program with ID 0 on NS 1
- C Read filtered data from subsystem memory to host

This presentation discusses NVMe work in progress, which is subject to change without notice.

 ©2022 Storage Networking Industry Association ©. Intel, Eideticom. All Rights Reserved.

Flow: Execute Program – Filter Encrypted Data



Flow steps

- A “Read” encrypted data into subsystem memory
- B Execute Program 1 on NS 1
- C Execute Program 0 on NS 1
- D Read filtered data from subsystem memory to host

This presentation discusses NVMe work in progress, which is subject to change without notice.

 ©2022 Storage Networking Industry Association ©. Intel, Eideticom. All Rights Reserved.

NVMe Changes for Computational Storage

Optional support

 ©2022 Storage Networking Industry Association ©. Intel, Eideticom. All Rights Reserved.

NVMe changes for Computational Storage

TP4091: Computational Programs

- New I/O command set for computational namespaces
- Commands:
  - Execute program
  - Load program
  - Activate program
  - Create/Delete Memory Range Set
- Support for Identify Controller, Namespace
- New log pages to support Computational Programs

TP4131: Subsystem Local Memory

- New I/O command set for memory namespaces
- Commands TBD
  - Commands to facilitate reading from and writing to memory namespaces
- Support for Identify Controller, Namespace
- New log pages TBD

 ©2022 Storage Networking Industry Association ©. Intel, Eideticom. All Rights Reserved.

NVMe Computational Storage Task Group

Join us!

 ©2022 Storage Networking Industry Association ©. Intel, Eideticom. All Rights Reserved.

Computational Storage Task Group

- **Task group co-chairs**
  - Kim Malone (Intel)
  - Stephen Bates (Eideticom)
  - Bill Martin (Samsung)
- **Task Group Goals**
  - Define the architecture of TP4091
  - Take TP4091 through to ratification
  - Other CS TPs
- **Membership**
  - 211 members from 47 companies
- **Join the task group**
  - https://workspace.nvmexpress.org/apps/org/workgroup/portal/
  - Select the Computational Storage Task Group
  - Click on the “Join Group” link
- **Task group meetings**
  - Thursdays 9 – 10 am Pacific time

 ©2022 Storage Networking Industry Association ©. Intel, Eideticom. All Rights Reserved.

Please take a moment to rate this session.

Your feedback is important to us.

 ©2022 Storage Networking Industry Association ©. Intel, Eideticom. All Rights Reserved.