DPDK expands into Storage domain

FIONA TRAHE, DAREK STOJACZYK

SEPTEMBER 2019

agenda

- What is SPDK?
- cryptodev
- compressdev
- memory management
- PCI access
- vhost
- Wrap-up

What is SPDK?

DPDK
DATA PLANE DEVELOPMENT KIT

Storage Performance Development Kit

Available via spdk.io

Open Source Software

- Optimized for latest generation CPUs and SSDs
- Software building blocks (BSD licensed)
- Designed to extract maximum performance from non-volatile media

Scalable and Efficient Software Ingredients

- User space, lockless, polled-mode components
- Up to millions of IOPS per core
- Minimize average and tail latencies

SPDK Community

Email Discussions

http://SPDK.IO

Main Web Presence

slack

Real Time Chat w/
Development Community

Weekly Calls
Multiple Annual Meetups

Trello

Backlog and
Ideas for Things to Do

GerritHub™

GitHub

Code Reviews & Repo

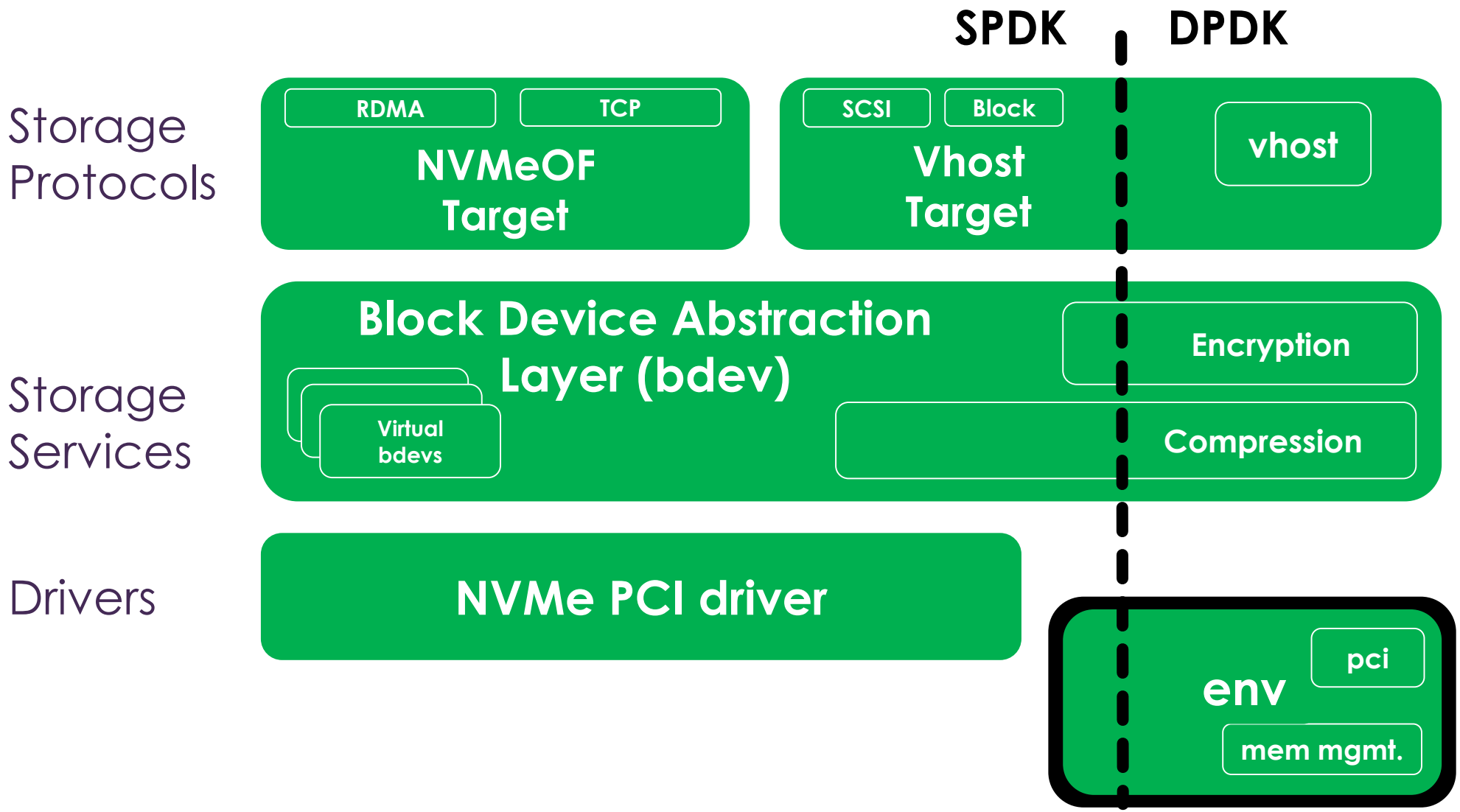
Jenkins

Continuous
Integration

SPDK HIGH-LEVEL ARCHITECTURE*

* stripped to only the modules relevant to this presentation

DPDK
DATA PLANE DEVELOPMENT KIT



agenda

- What is SPDK?
- cryptodev
- compressdev
- memory management
- PCI access
- vhost
- Wrap-up

CRYPTO BDEV MODULE

DPDK
DATA PLANE DEVELOPMENT KIT

Application

<SPDK Defined Block API>

Block Device Layer

VBDEV

VBDEV | Crypto

BDEV | BDEV

<Driver Specific API>

Drivers

DPDK

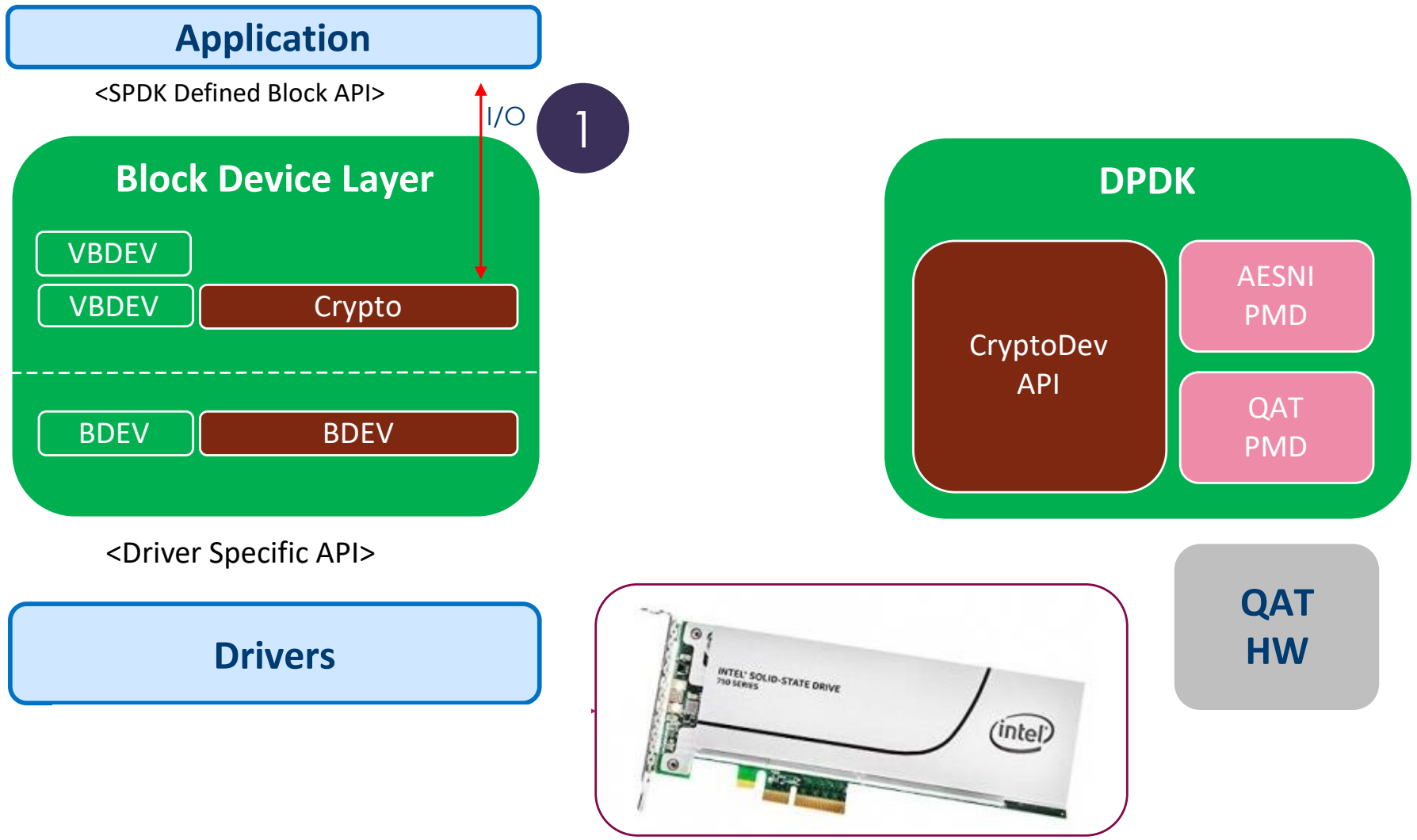
CryptoDev API

AESNI PMD

QAT PMD

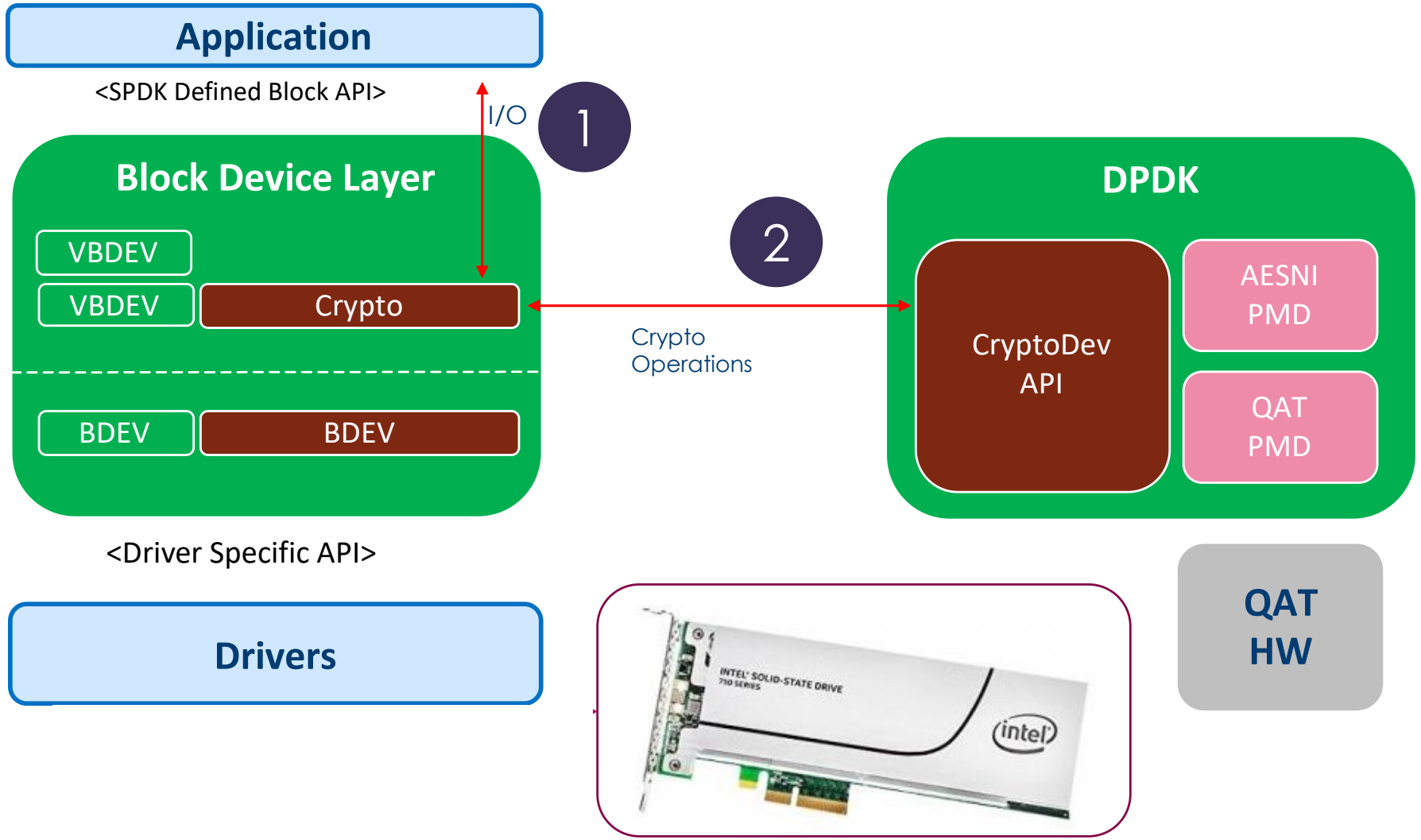
QAT HW

CRYPTO BDEV MODULE



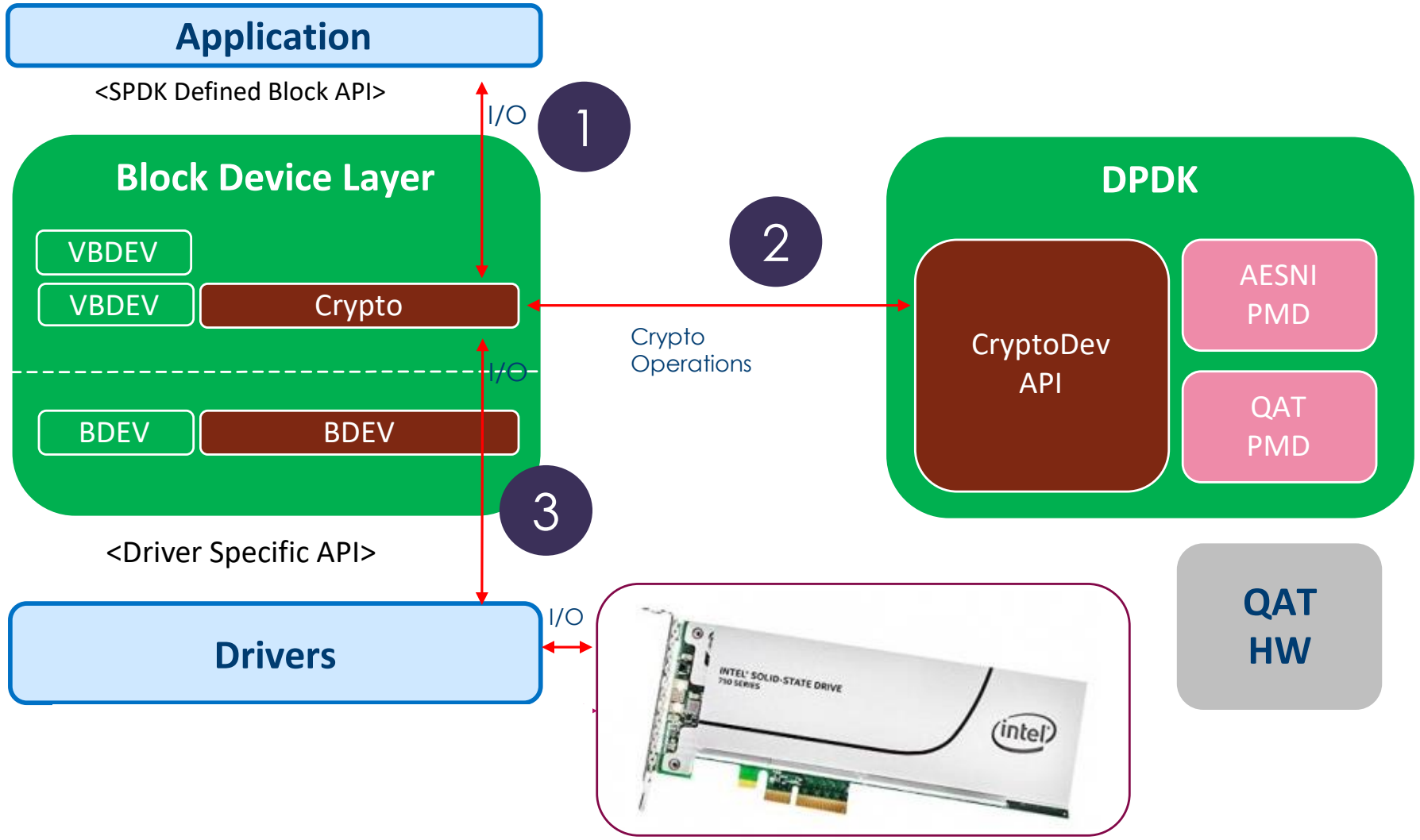
CRYPTO BDEV MODULE

DPDK
DATA PLANE DEVELOPMENT KIT



9

CRYPTO BDEV MODULE



Cryptodev integration

- Uses cipher-only. No auth, no cipher-auth chaining.
- Cipher algorithm: AES-CBC
- AES-XTS – an algorithm commonly used for disk encryption has been implemented in cryptodev. On SPDK backlog to be integrated.
- Uses QAT PMD for hardware encryption and aesni-mb PMD for software encryption

agenda

- What is SPDK?
- cryptodev
- compressdev
- memory management
- PCI access
- vhost
- Wrap-up

COMPRESSION BDEV MODULE

DPDK
DATA PLANE DEVELOPMENT KIT

Application

<SPDK Defined Block API>

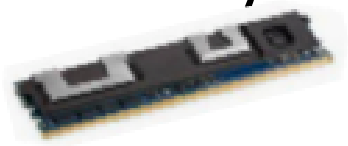
Block Device Layer

- VBDEV
- VBDEV
- libReduce
- PMDK
- Compression
- BDEV
- BDEV

<Driver Specific API>

Drivers

Persistent Memory



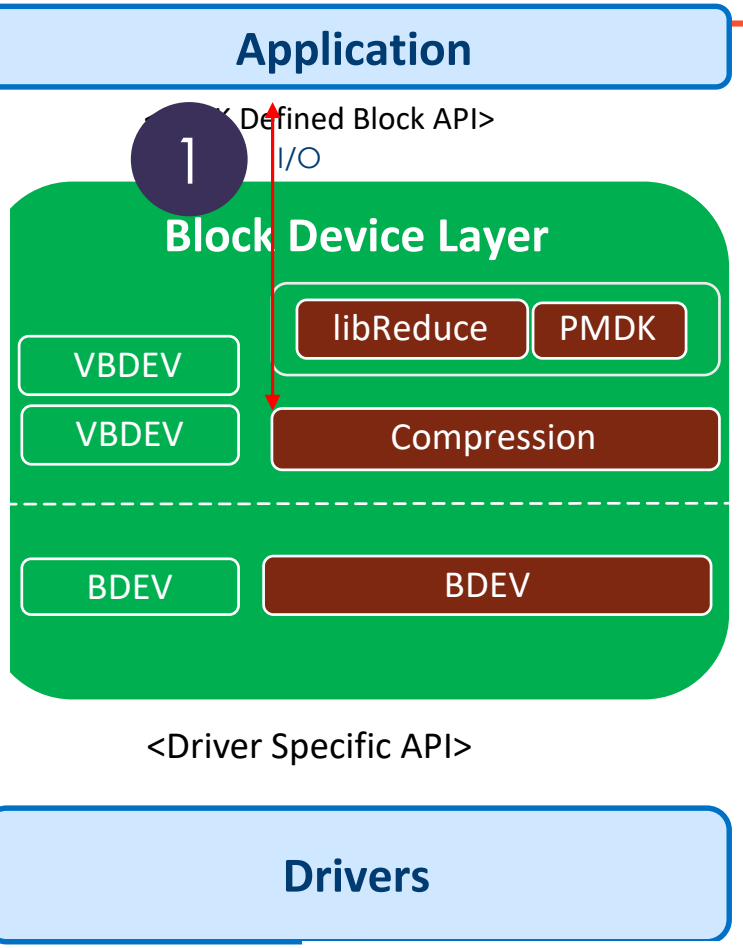
DPDK

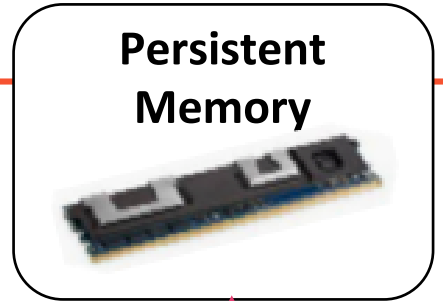
- compressdev API
- ISA-L PMD
- QAT PMD

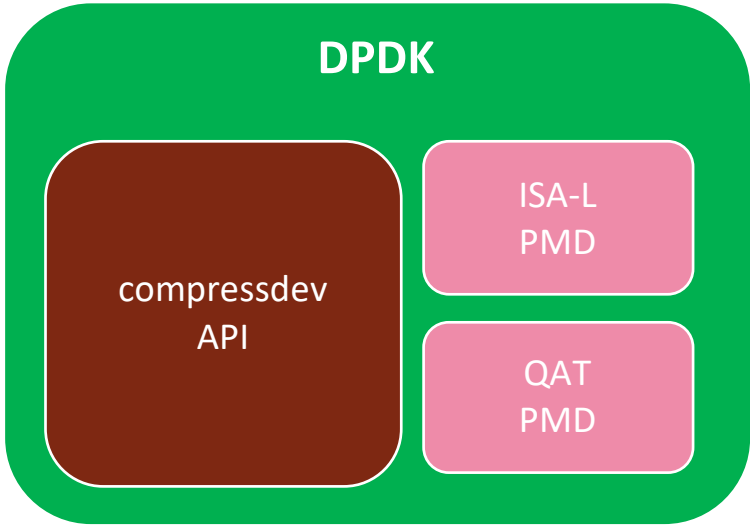
QAT HW

COMPRESSION BDEV MODULE

DPDK
DATA PLANE DEVELOPMENT KIT



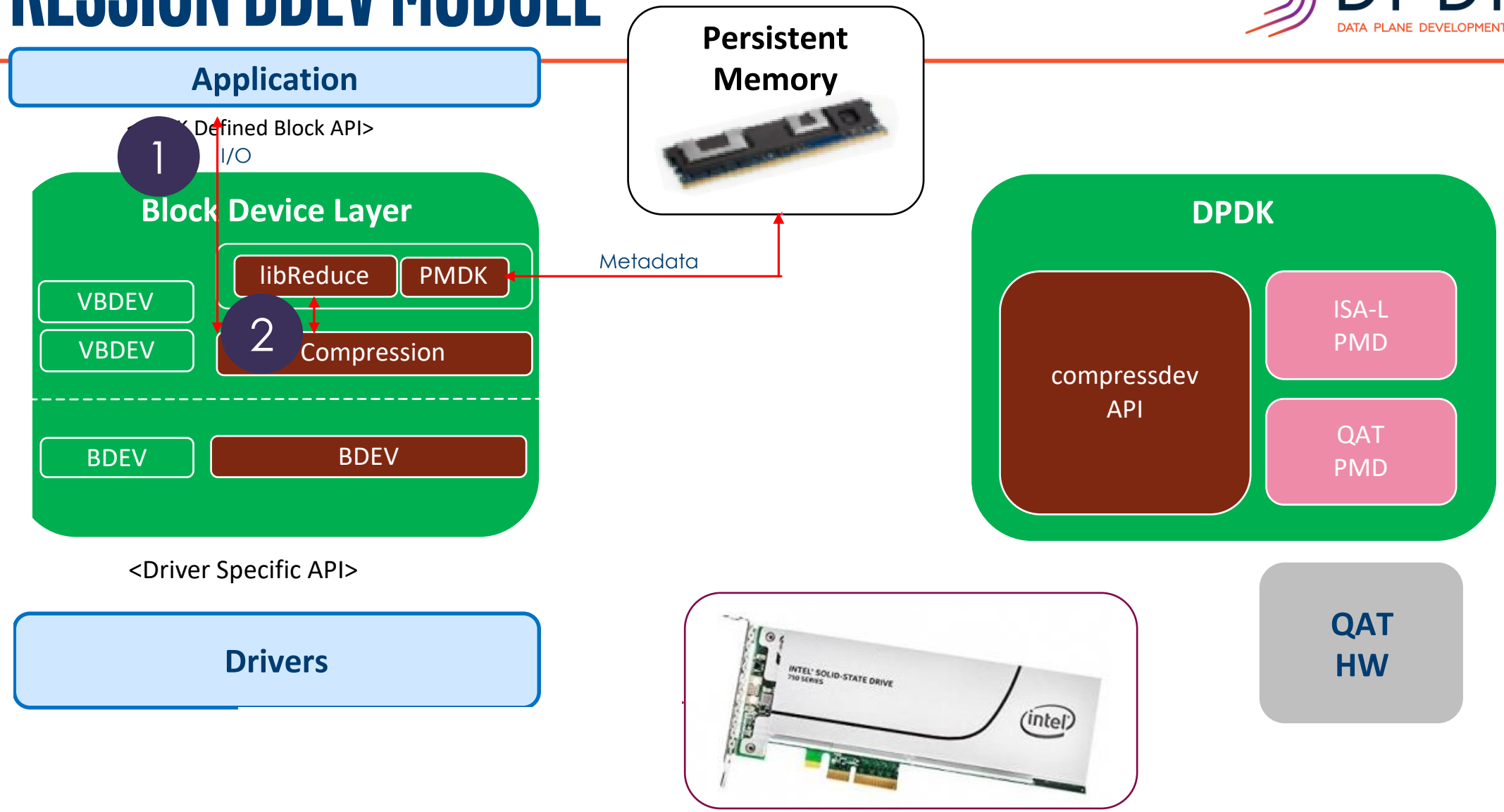




QAT
HW

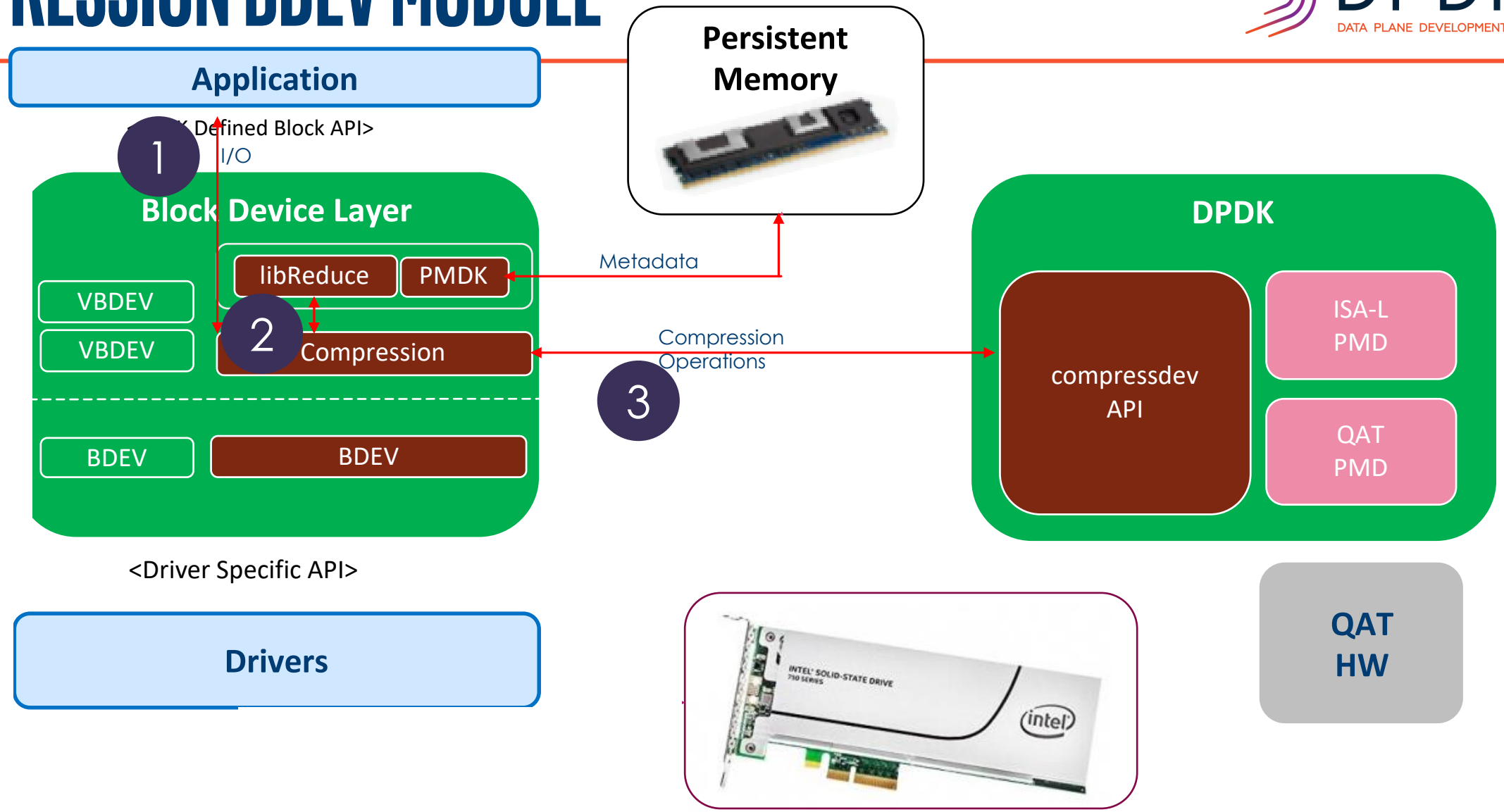
COMPRESSION BDEV MODULE

DPDK
DATA PLANE DEVELOPMENT KIT



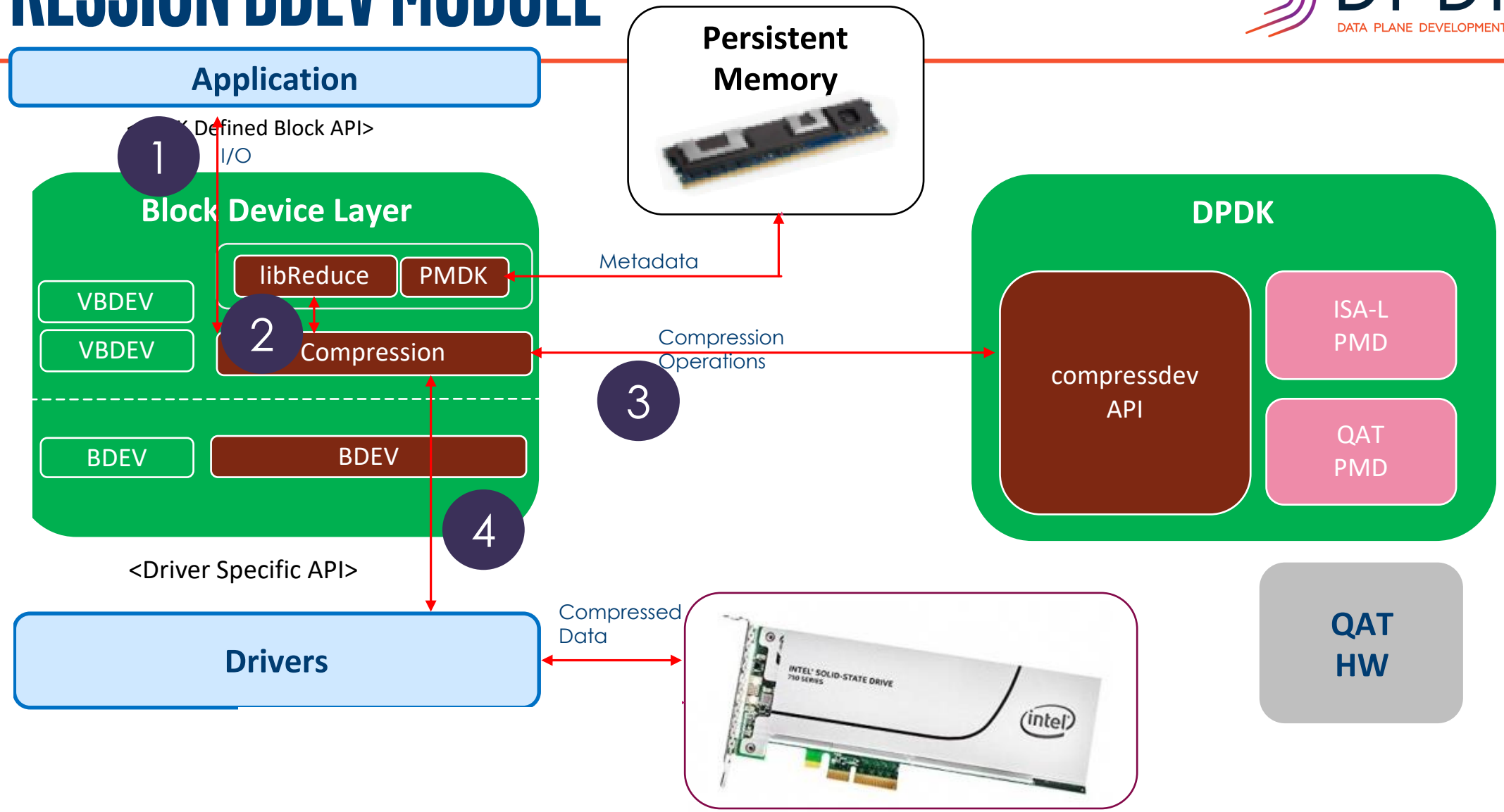
COMPRESSION BDEV MODULE

DPDK
DATA PLANE DEVELOPMENT KIT



COMPRESSION BDEV MODULE

DPDK
DATA PLANE DEVELOPMENT KIT



Compressdev integration

- More complicated than crypto
- Output data size is different to input data size and unpredictable.
- However one advantage over network compression use-cases is that the decompressed data size is known
- libReduce figures out which LBAs in the application i/f map to which LBAs on the backing device
- It uses persistent memory (PMDK) to store meta-data.
- QAT and ISAL PMDs are used for compression & decompression

agenda

- What is SPDK?
- cryptodev
- compressdev
- memory management
- PCI access
- vhost
- Wrap-up

Memory management

- spdk_malloc()
- spdk_mem_register()

Memory management

- spdk_malloc() → calls rte_malloc()
- spdk_mem_register()

Memory management

- spdk_malloc() → calls rte_malloc()
- spdk_mem_register() → calls rte_vfio_dma_map()

Memory management

- spdk_malloc() → calls rte_malloc()
- spdk_mem_register() → calls rte_vfio_dma_map()

… also calls ibverbs APIs

Memory management

- spdk_malloc() → calls rte_malloc()
- spdk_mem_register() → calls rte_vfio_dma_map()

… also calls ibverbs APIs

… also shares the memory
with connected vhost devices

agenda

- What is SPDK?
- cryptodev
- compressdev
- memory management
- PCI access
- vhost
- Wrap-up

PCI access

- spdk_pci_device_attach()
- spdk_pci_device_detach()

PCI access

- spdk_pci_device_attach() → can call rte_eal_hotplug_add()
  returns spdk_pci_device *

- spdk_pci_device_detach()

PCI access

- spdk_pci_device_attach() → can call rte_eal_hotplug_add()
  returns spdk_pci_device *
- spdk_pci_device_detach() → calls rte_eal_hotplug_remove()

PCI access

- spdk_pci_device_attach() → can call rte_eal_hotplug_add()
  returns spdk_pci_device *
- spdk_pci_device_detach() → calls rte_eal_hotplug_remove()
- rte_dev_event_callback_register()

PCI access

- spdk_pci_device_attach() → can call rte_eal_hotplug_add()
  returns spdk_pci_device *
- spdk_pci_device_detach() → calls rte_eal_hotplug_remove()
- rte_dev_event_callback_register()

  → sets spdk_pci_device->removed = 1

agenda

- What is SPDK?
- cryptodev
- compressdev
- memory management
- PCI access
- vhost
- Wrap-up

Vhost

- DPDK's library for creating and polling vhost devices
- Originally created for vhost-net
- Implements mostly device-agnostic vhost-user protocol
- SPDK uses it for storage

DPDK 19.05 solution

rte_vhost_extern_callback_register()

- Hooks a function to be called on each vhost message.
- Allows overriding default rte_vhost message handling

agenda

- What is SPDK?
- cryptodev
- compressdev
- memory management
- PCI access
- vhost
- Wrap-up

Questions?

Fast-track to relevant SPDK urls

- The SPDK project is here: spdk.io
- SPDK codebase: github.com/spdk/spdk
- Docs describing crypto & compression bdevs and compression design

  spdk.io/doc/bdev.html

  spdk.io/doc/reduce.html