LINUXCON containercon CHINA 中国 CLOUDOPEN

THINK OPEN
开放性思维

Intel® Scalable I/O Virtualization

Kevin Tian

Principal Engineer, Intel

LF ASIA, LLC

Legal Disclaimer

No license (express or implied, by estoppel or otherwise) to any intellectual property rights is granted by this document.

Intel disclaims all express and implied warranties, including without limitation, the implied warranties of merchantability, fitness for a particular purpose, and non-infringement, as well as any warranty arising from course of performance, course of dealing, or usage in trade.

This document contains information on products, services and/or processes in development. All information provided here is subject to change without notice. Contact your Intel representative to obtain the latest forecast, schedule, specifications and roadmaps.

The products and services described may contain defects or errors known as errata which may cause deviations from published specifications. Current characterized errata are available on request.

Copies of documents which have an order number and are referenced in this document may be obtained by calling 1-800-548-4725 or by visiting www.intel.com/design/literature.htm.

Intel and the Intel logo are trademarks of Intel Corporation or its subsidiaries in the U.S. and/or other countries.

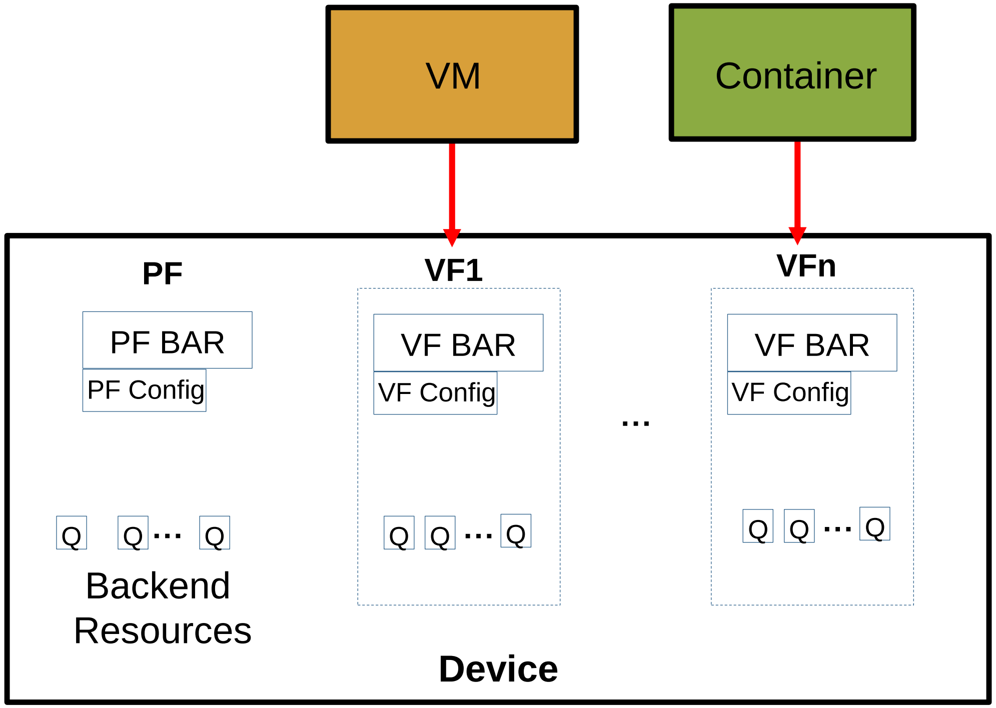
*Other names and brands may be claimed as the property of others

Hardware-Assisted I/O Virtualization

- Pursued for two classes of devices
  - High-performance devices where SW method imposes large overhead
    - E.g. NICs, RDMA devices, NVMe, etc.
  - Complex devices where virtualizing the device entirely in software is not practical
    - E.g. GPU, FPGA, etc.

- Today SR-IOV is the standard framework for PCI Express® devices

PCI Express® SR-IOV



- PCIe® Single Root I/O Virtualization (SR-IOV)
  - ✓ Physical Function (PF)
  - ✓ Virtual Function (VF)
- VF directly assignable to
  - ✓ Traditional Virtual Machine (VM)
  - ✓ Bare metal container/process
  - ✓ VM container

New Requirements

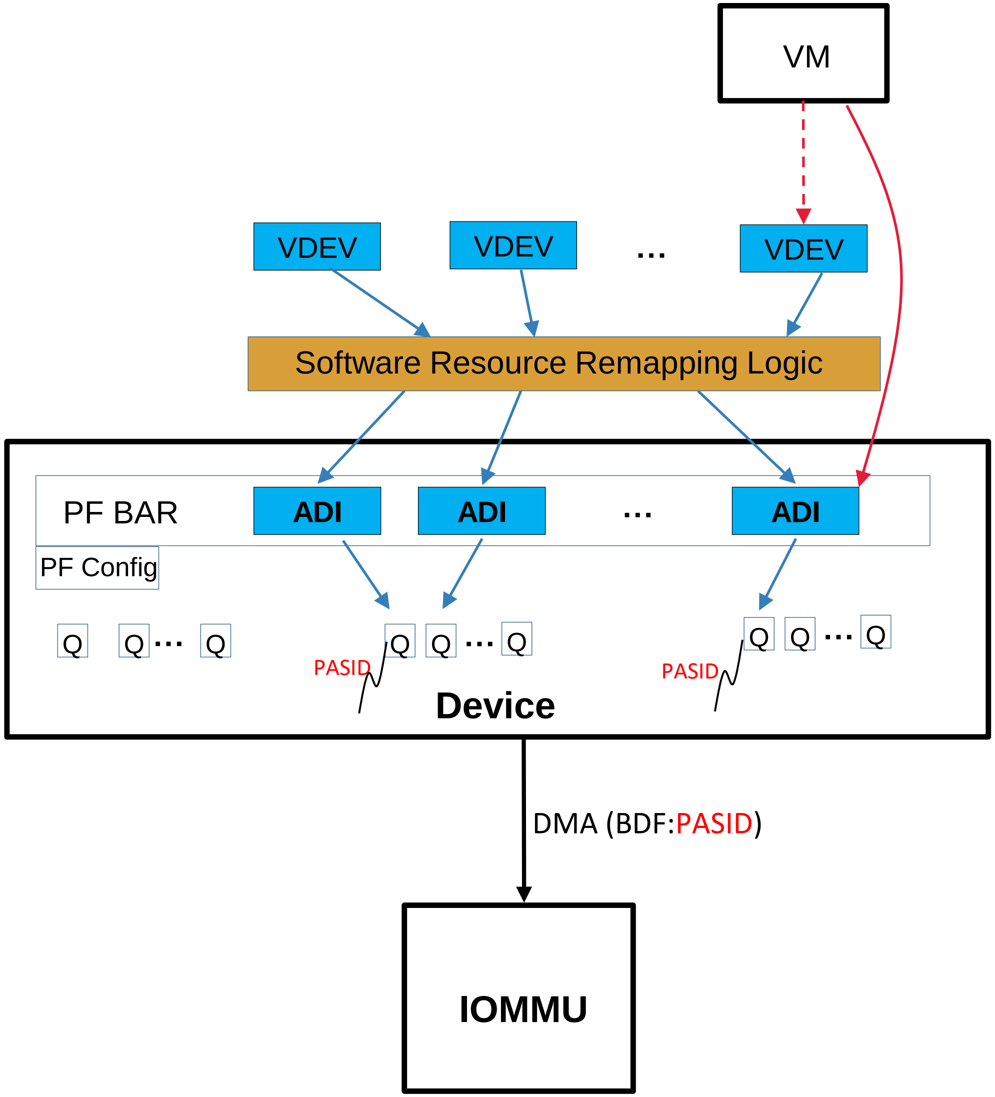
- Hyper-scale environment
  - Scale to 1000+ VMs/containers
- Dynamic resource management
  - User-defined sharing granularity, over-provisioning, etc.
- Composability
  - VM live migration, snapshot, generational compatibility, etc.

Observed major limitations on SR-IOV!

Intel® Scalable I/O Virtualization (Intel® Scalable IOV)

- A hardware-assisted mediated pass-through architecture
  - Slow-path operations emulated by software
  - Fast-path resources dynamically provisioned for direct access
  - Hardware-enforced DMA isolation between fast-path resources
- Finer-grained device sharing than SR-IOV
  - Think about each TX/RX queue pair is now assignable
- Utilizes existing PCIe® capabilities
  - e.g. Process Address Space ID (PASID)
- Supports any type of devices
  - e.g. NIC, storage, GPU, accelerators, … (integrated or discrete)
- Supports both VM and bare-metal usages

Intel® Scalable IOV Concept



■Device: Assignable Device Interfaces (ADI)

- ✓ Queues, queue pairs, contexts
- ✓ Meet isolation criteria to be ‘assignable’
- ✓ Tagged with unique PASID

■Platform: PASID-granular DMA isolation

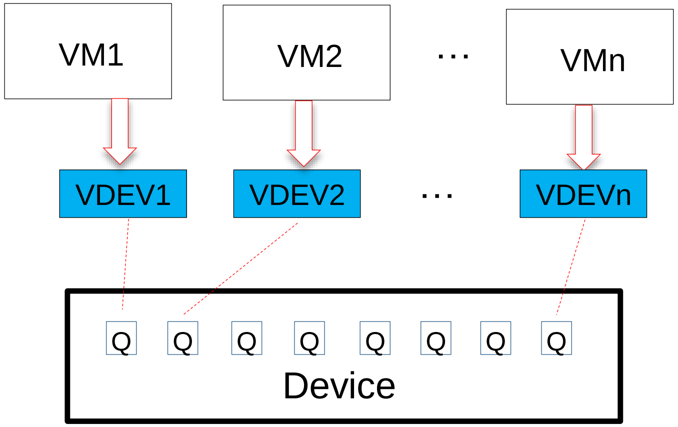
- ✓ Through Intel® VT-d extensions

■Software: Compose ADIs into Virtual Device (VDEV)

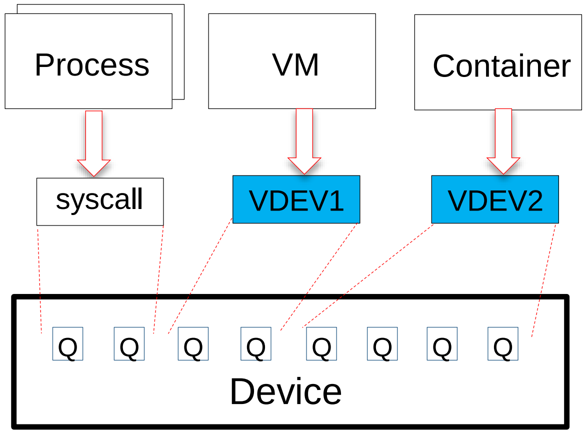
- ✓ Software managed resource remapping between VDEV and ADI
- ✓ Slow-path emulation & fast-path pass-through

Benefits

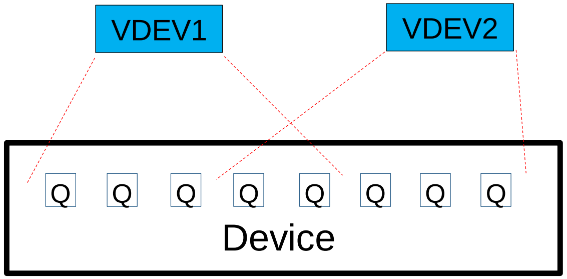
Scalability



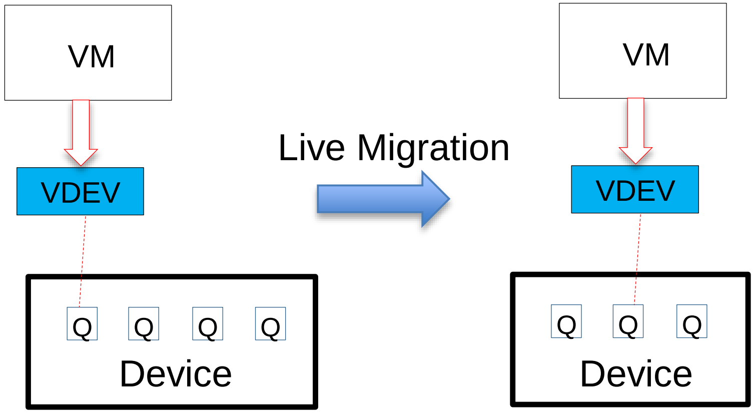
Flexibility



Over-provisioning



Composability



Assignable Device Interfaces (ADIs)

- Smallest granularity of sharing a device
  - No PCI config space register, share common BDF
  - Identified by PASID

- For ADI to be ‘assignable’
  - Functional isolation between ADIs
  - ADI MMIO registers in separate system page size regions
  - All DMAs tagged with PASID
  - Independently resettable
  - Scalable Interrupt Message Storage (IMS)
  - …

Enumeration of Intel® Scalable IOV Capability

- Designated Vendor Specific Extended Capability (DVSEC) to discover Intel® Scalable IOV capability
  - A simplified subset of SR-IOV capability

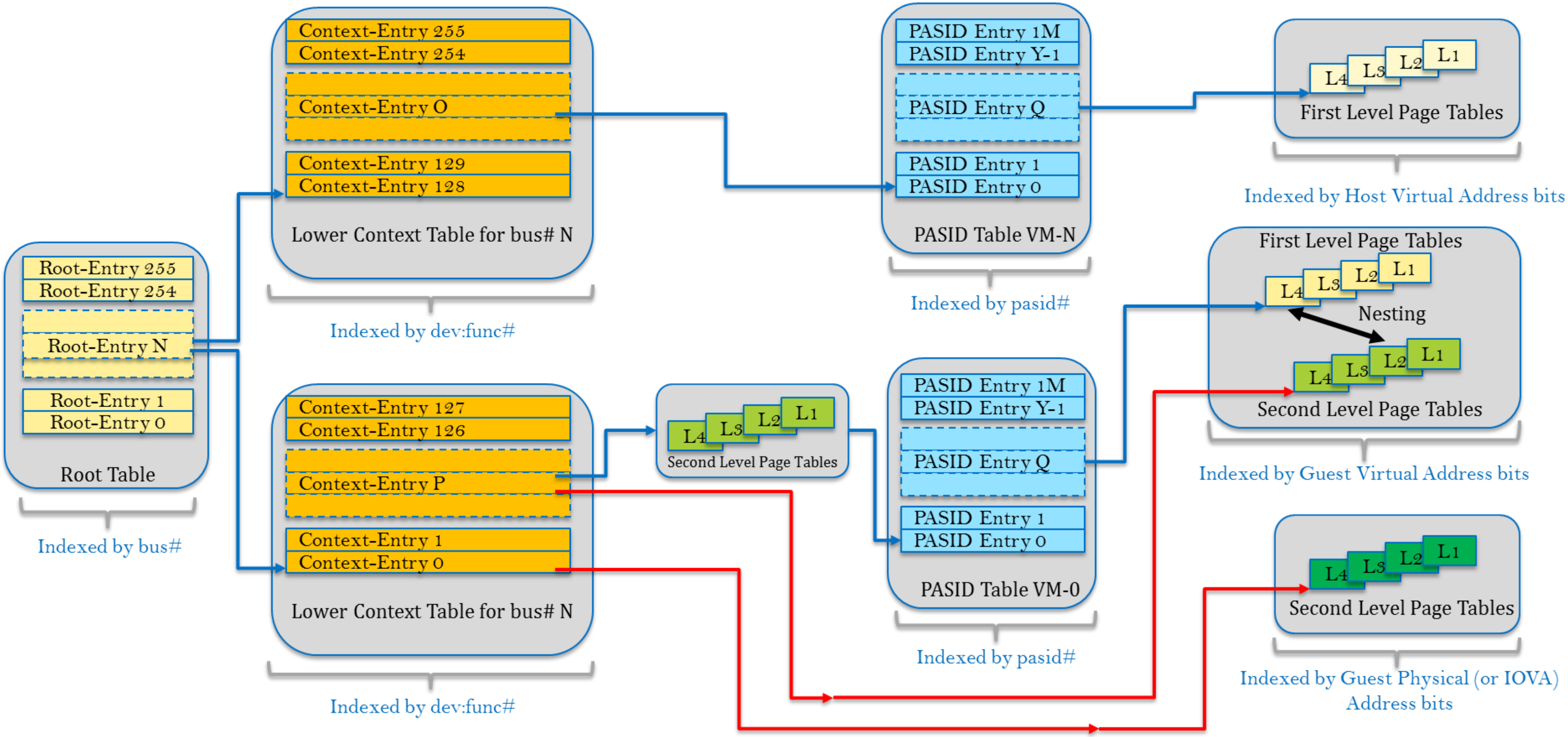
31 24	23 20	19 16	15 0	Byte Offset
Next Capability Offset		Cap Version	PCI Express Extended Capability ID = 0x23	00h
DVSEC Length = 0x18		DVSEC rev = 0	DVSEC Vendor ID = 8086	04h
Flags (RO)	Function Dependency Link (RO)		DVSEC ID for Scalable IOV = XXX	08h
Supported Page Sizes (RO)				0Ch
System Page Size (RW)				10h
Capabilities (RO)				14h

Intel® VT-d Enhancement

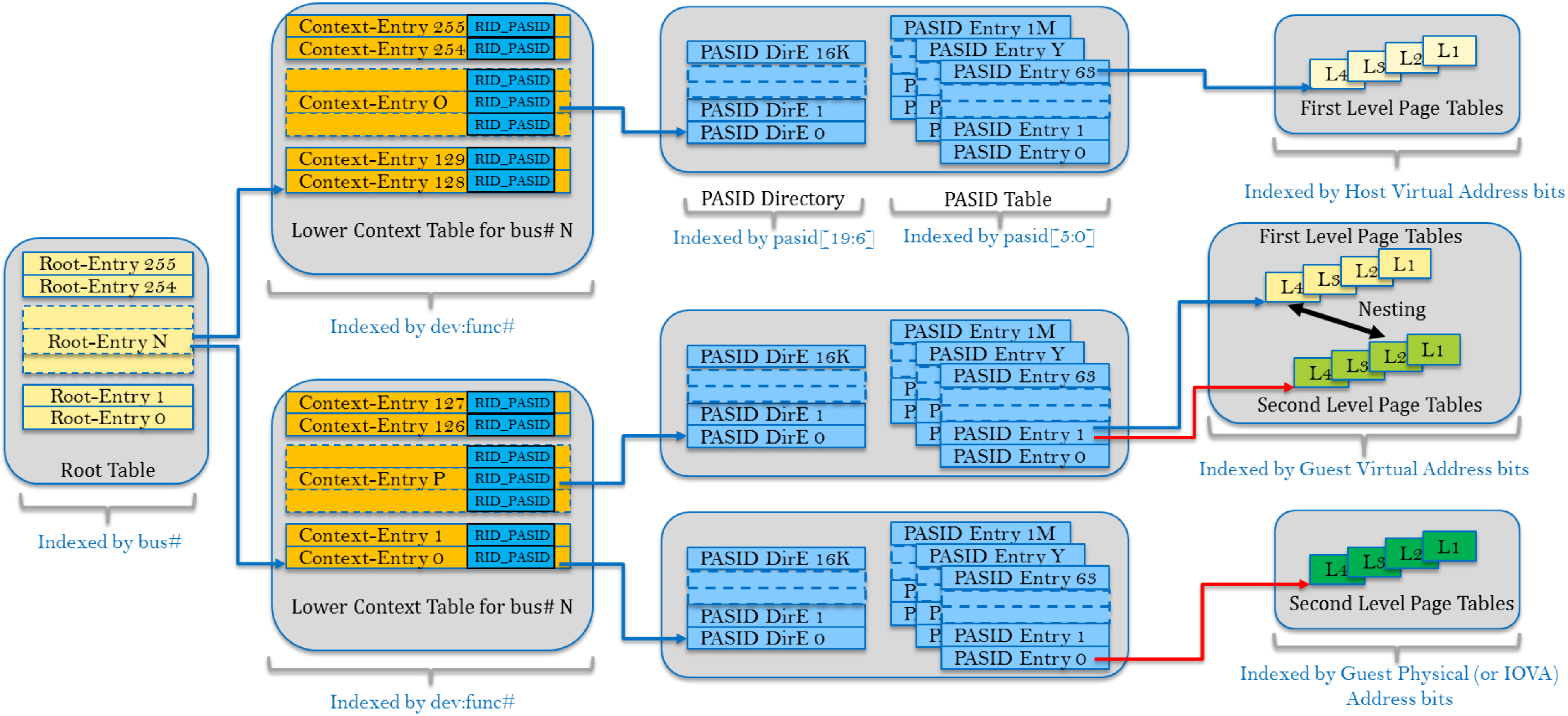
- Scalable mode DMA remapping
  - PASID granule 1st-level, 2nd-level, nested and pass-through
  - PASID table now two-level structure
  - Cover both Scalable IOV and SVM usages
    - Extended Context (ECS) is deprecated

- Access/Dirty (A/D) bits in 2nd-level
  - Assist dirty memory tracking in live migration

Extended Context Mode (Deprecated)



Scalable Mode (New)

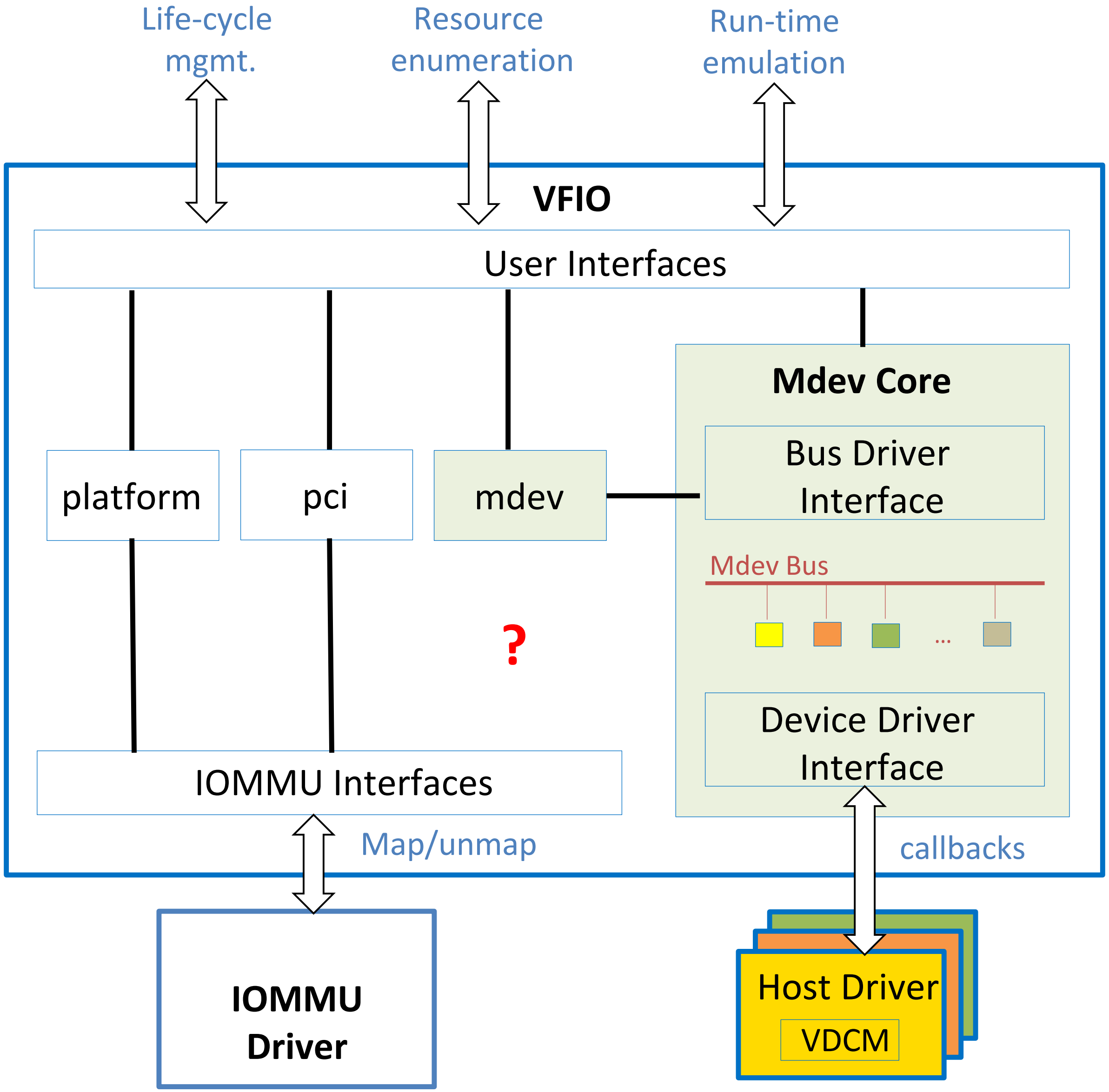


<u>Key Difference</u>: PASID is a global ID space shared by all VMs.
ALL page-table pointers moved to PASID Granular table

Software Composition

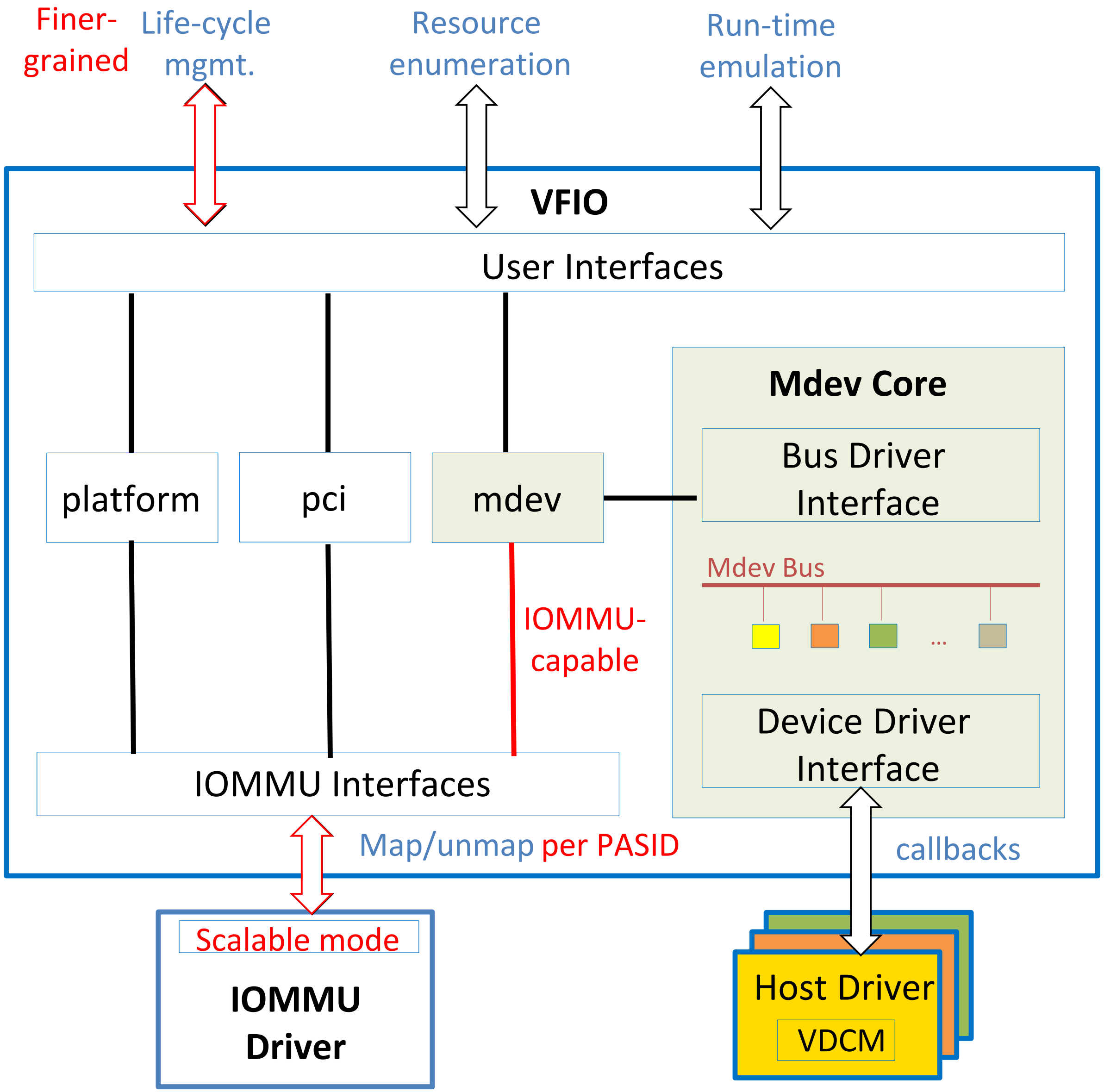
- Virtual Device Composition Module (VDCM)
  - Compose ADIs into Virtual Device (VDEV)
  - Emulate slow-path operations
- Need a framework to connect VDCM for
  - Managing VDEV life-cycle
  - Setting up access policy on VDEV resources
  - Serving slow-path operations from guest
- In Linux it’s VFIO mediated device framework!
  - “mdev” == “VDEV” in concept

VFIO Mediated Device Framework



- Mdev core
  - ✓ Connect VFIO and VDCM
- User interfaces
  - ✓ Used by libvirt, qemu, etc.
- IOMMU map/unmap
- DMA isolation for mdev
  - ✓ Purely in software, or
  - ✓ In vendor specific way

Extensions for Intel® Scalable IOV



■IOMMU-capable mdev

- ✓ Link to iommu_domain (tagged by PASID)
- ✓ Allow PASID-granular iommu map/unmap
- ✓ Opt-in by VDCM

■Finer-grained resource management

- ✓ Specify any number of ADIs to compose a mdev

■Unified framework for VM and bare metal usages

- ✓ Mdev composition can be usage specific, e.g. no PCI emulation in bare metal usage

Main Linux Enabling Tasks

- To enable basic ADI assignment
  - Support new scalable mode
  - Need system-wide PASID space
  - Introduce iommu-capable mdev
  - Device specific VDCM in host driver

- To support vIOMMU/vSVM with ADI
  - Emulate new scalable mode on vIOMMU
  - Enlightened PASID management scheme
  - Maintain compatible APIs between PF/VF and ADI

Summary of Architecture Changes

Device Support

- Support Assignable Device Interfaces (ADIs)
- Support direct fast-path access from VMs

Platform Support

- Extend Intel® VT-d to use PASID/BDF to identify DMA upstream accesses

Software Support

- Move infrequent (slow-path) accesses from the device to software without affecting perf

Documentation

- Intel® VT-d specification update (Rev 3.0)
  - Documents Intel® VT-d (IOMMU) support for PASID granular address translation
- Intel® Scalable I/O Virtualization Technical Specification (Rev 1.0)
  - Documents the Scalable IOV architecture blueprint and operation, including DVSEC
  - Addresses architecture requirements for devices and drivers
  - Agnostic of type of device or specific implementation
  - Openly published to enable broad device and software ecosystem
- https://software.intel.com/en-us/articles/intel-sdm

LINUXCON

containercon

CLOUDOPEN

CHINA 中国

THINK OPEN

开放性思维

Q/A

LF ASIA, LLC

LINUXCON

containercon

CLOUDOPEN

CHINA 中国

THINK OPEN

开放性思维

LF ASIA, LLC