



IOMMUFD SUPPORT FOR MICROSOFT HYPERVISOR HOSTS

MICROSOFT/LINUX SYSTEM GROUP
JACOB PAN <JACOB.PAN@LINUX.MICROSOFT.COM>



LINUX
PLUMBERS CONFERENCE

PRAGUE, CZECHIA / OCT. 5-7, 2026



ACKNOWLEDGMENTS

Jason Gunthorpe — For reviewing our earlier code and suggesting the anchor vIOMMU concept

Mukesh Rathor & Wei Liu — For helping me understand the MSHV architecture

John Starks — For sharing insights into the virtualization stack and openVMM

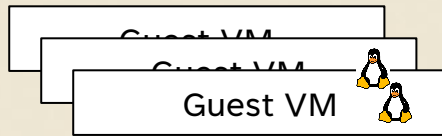


LINUX
PLUMBERS CONFERENCE

PRAGUE, CZECHIA / OCT. 5-7, 2026



WHY DO WE CARE?

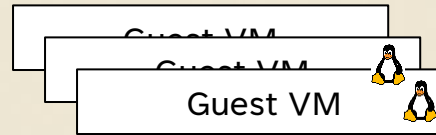


Hyper-V

vmwp.exe — Virtual Machine Worker Process

Windows Host

Microsoft Hypervisor



Opensource VMMs (e.g. openVMM)

Linux Host

Microsoft Hypervisor

IOMMU patches

status

Guest PV IOMMU driver

v5

Host PV IOMMU driver

v2

IOMMUFD extensions for MSHV

RFC

PASID + SVA support

WIP



LINUX
PLUMBERS CONFERENCE

PRAGUE, CZECHIA / OCT. 5-7, 2026

OVERVIEW

- **Microsoft Hypervisor(MSHV) device assignment architecture (relevance to Xen)**
 - Structural differences: MSHV vs KVM
 - Problem statement
- **Proposed IOMMUFD extensions**
- **Open Discussion**



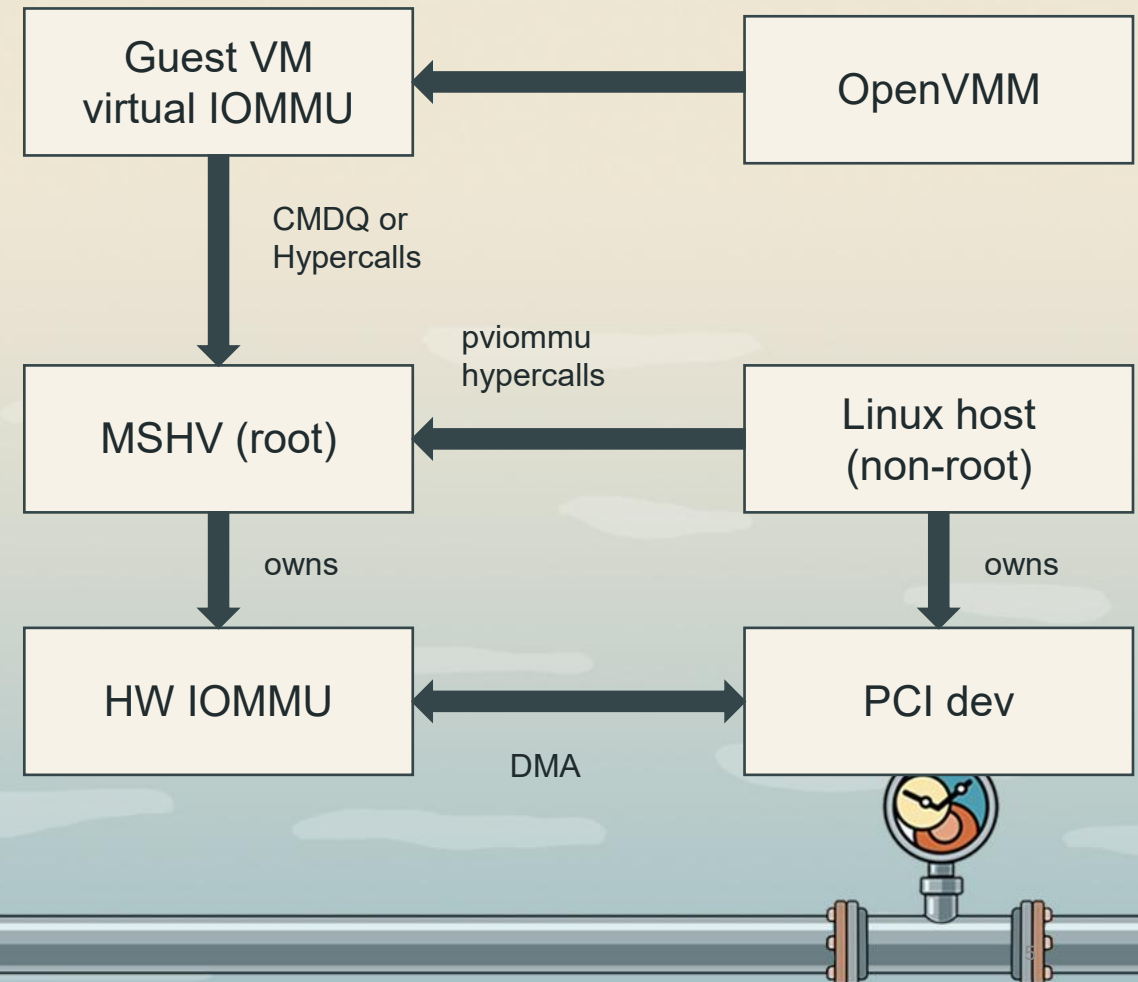
LINUX
PLUMBERS CONFERENCE

PRAGUE, CZECHIA / OCT. 5-7, 2026



DEVICE ASSIGNMENT ARCHITECTURE

1. Linux root partition (host OS) owns the PCI devices
2. Microsoft Hypervisor (MSHV) owns physical IOMMU
3. Guest virtual IOMMU emulation is done in MSHV rather than the user VMM
4. S2 IOPT may be shared with CPU SLAT
5. Linux host programs HW IOMMU S2 via PVIOMMU driver by map/unmap hypercalls (**expensive!**)



TWO HOST DEVICE-DOMAIN ATTACHMENT MODELS

1. Attach device to nesting parent S2 domain established within IOMMUFD (Existing)
2. Attach device to hypervisor managed S2, external to host IOMMUFD (**New**)



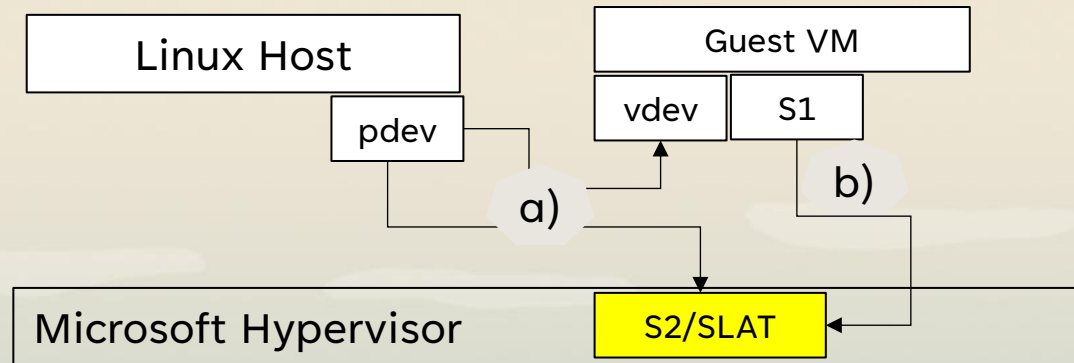
LINUX
PLUMBERS CONFERENCE

PRAGUE, CZECHIA / OCT. 5-7, 2026



IOMMUFD: INTRODUCE EXTERNAL ATTACH

Definition: Move a device to a guest with an S2 IOPT managed by the hypervisor, external to the host IOMMUFD.



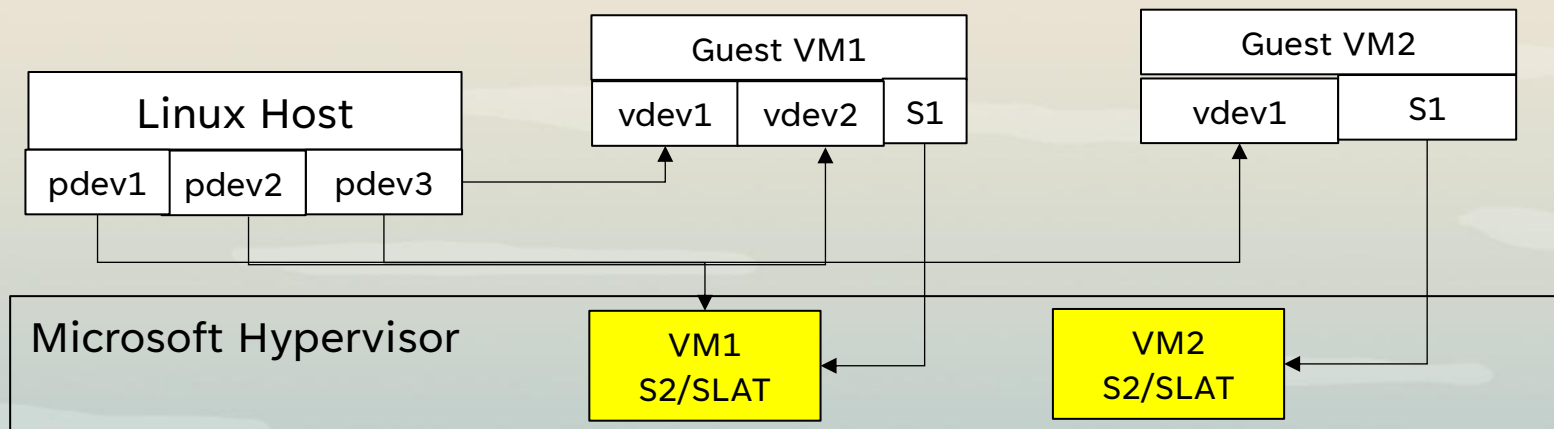
Nested translation attachment is decoupled :

- a) The host attaches the device to S2 via a hypercall
- b) The guest configures S1 and establishes nesting directly with MSHV.



WHAT EXTERNAL ATTACH NEEDS FROM IOMMUFD & DRIVER?

1. The host pVIO MMU driver needs VM context besides pdevice ID:
 - The target guest partition identity.
 - The device's guest-visible logical identity
2. Lifetime alignment between partition and attached devices
3. Coordination for guest-memory pinning/mapping between MSHV and IOMMUFD



SOLUTION: PROPOSAL FOR NEW IOMMUFD OBJECTS

New! IOMMU_DOMAIN_EXTERNAL

- A non-paging domain managed by an external owner

New! IOMMUFD_VIOMMU_TYPE_HYPERVISOR

- A per-VM object that holds a reference to the VM/partition file to anchor its lifetime.
- Type-specific helpers resolve the partition ID and coordinate guest-memory pinning and mapping.

New! IOMMUFD_OBJ_HWPT_EXTERNAL - IOMMUFD obj for iommu external domain

Reuse IOMMUFD_VDEVICE - Associates the physical device with its guest-visible logical device ID.

RFC PATCH <https://lore.kernel.org/linux-iommu/20260925190742.1575380-1-jacob.pan@linux.microsoft.com/>

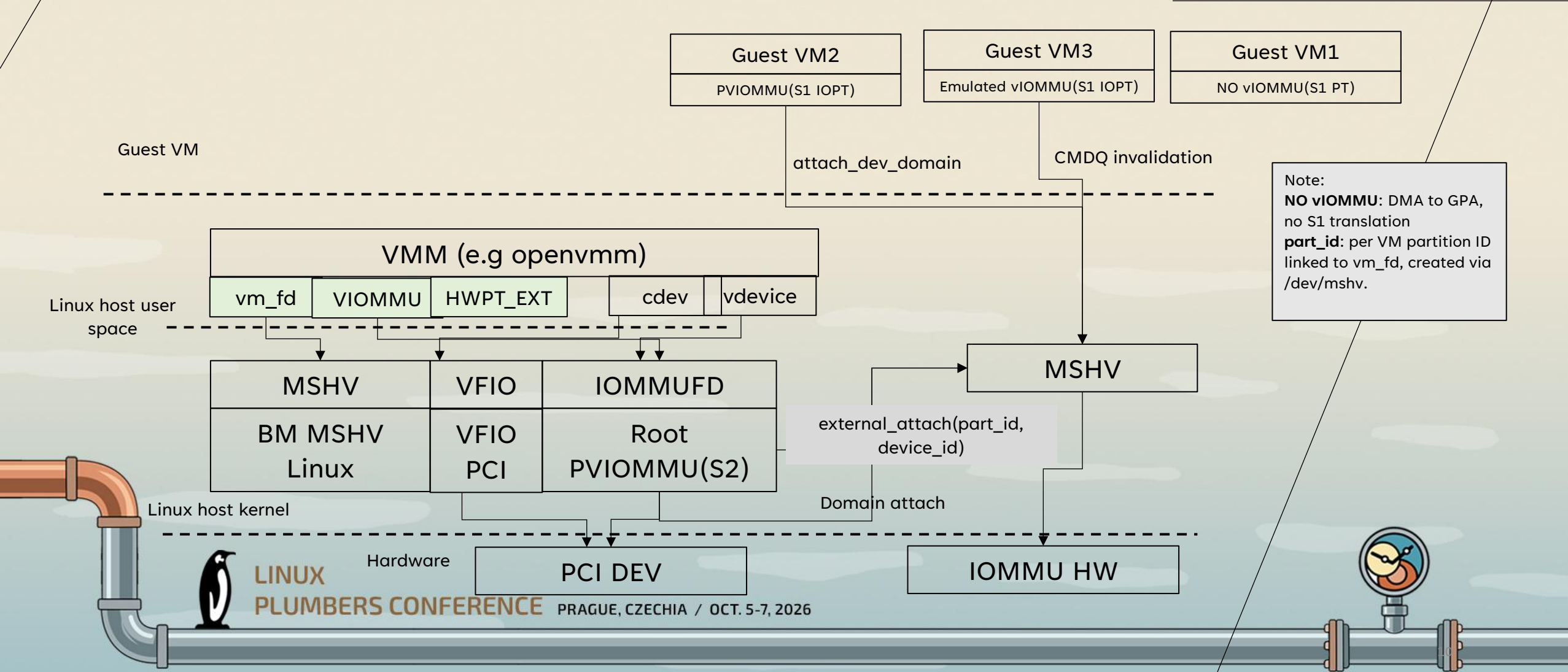


LINUX
PLUMBERS CONFERENCE

PRAGUE, CZECHIA / OCT. 5-7, 2026



DEVICE ASSIGNMENT MODEL: LINUX HOST



USERSPACE VMM FLOW FOR EXTERNAL ATTACH

1. `iommufd = open("/dev/iommu")`
2. `vfio_fd = open("/dev/vfio/devices/vfioN")`
3. `dev_id = bind_vfio_device(vfio_fd, iommufd) // VFIO_DEVICE_BIND_IOMMUFD`
4. `create_mshv_partition(&mshv_fd, &vm_fd, flags)`
5. `alloc_hypervisor_viommu(iommufd, &viommu_id, dev_id, vm_fd) /* IOMMUFD_VIOMMU_TYPE_HYPERVISOR */`
6. `alloc_vdevice(iommufd, viommu_id, dev_id, vm_device_id, &vdevice_id) /* vm_device_id is assigned by VMM */`
7. `alloc_external_hwpt(iommufd, dev_id, viommu_id, &external_hwpt_id)`
8. `attach_external_hwpt(vfio_fd, external_hwpt_id) /* VFIO_DEVICE_ATTACH_IOMMUFD_PT */`
 - > MSHV root pvIOMMU driver invokes external iommu domain attach_device op
 - > resolve partition ID from hypervisor viommu
 - > resolve logical device ID from vDEVICE
 - > invoke the partition prepare_attach callback to force pinning if policy requires
 - > MSHV attach-device hypercall



WIP

- PASID/ATS enabling contract in host attach & guest attach is WIP
 - PASID & ATS usage is unknown to the host during external attach
- L1VH Level-1 Virtual Host, a semi privileged L1 VM that can create and manage guest VMs



LINUX
PLUMBERS CONFERENCE

PRAGUE, CZECHIA / OCT. 5-7, 2026



OPEN DISCUSSIONS

- Would the external attach flow work for Xen? Share vIOMMU helpers with TSM
 - <https://lore.kernel.org/linux-iommu/20260925124529.GJ9354@nvidia.com/T/#t>
- Live update, should /dev/mshv register with LUO sessions to preserve mapping?
- Revokable dma-buf support for P2PDMA, should /dev/mshv be the importer?
- Any unknown unknowns?



LINUX
PLUMBERS CONFERENCE

PRAGUE, CZECHIA / OCT. 5-7, 2026



BACKUPS:TERMINOLOGY

- **VM (virtual machine) aka partition:** Guest operating system (linux/windows/BSD..) running on a hypervisor.
- **MSHV aka Microsoft Hypervisor.**
- **Hyper-V,** Microsoft Hypervisor plus Windows software stack
- **Baremetal linux:** this means there is NO hypervisor, just linux running natively on hw.
- **Baremetal Linux root partition:** linux running as the host partition on hypervisor. Aka linux host.
- **L1 VM/guest:** VM/guest started by baremetal dom0 or L1 Linux root partition.
- **l1vh or l1vh:** Level-1 Virtual Host, a semi privileged L1 VM that can create and manage guest VMs.
- **guest VM:** normally, a VM started by BM root partition or l1vh or L2 root partition ... ie, an unprivileged VM.
- **VMM:** user space part of virtualization component, such as qemu/cloud hypervisor/openvmm, that runs in linux L1 root partition or linux l1vh or linux L2 root partition.



LINUX
PLUMBERS CONFERENCE

PRAGUE, CZECHIA / OCT. 5-7, 2026



EXTERNAL ATTACH VS. IOAS NESTED DOMAINS ATTACH

MSHV/Hyper-V

Host external attach

IOMMUFD VIOMMU + vDEVICE

-> HWPT_EXTERNAL -> device attach

S2 shared SLAT

v

Guest IOMMU programming (later)

emulated/pvIOMMU -> guest S1

v

MSHV forms guest S1 + S2

v

Nested DMA translation

Nested domains(all in IOMMUFD)

Host S2 setup

IOAS -> HWPT_PAGING (parent)

v

Guest IOMMU programming (later)

emulated IOMMU -> guest S1

v

QEMU allocates HWPT_NESTED

parent = S2, data = S1

v

Nested DMA translation

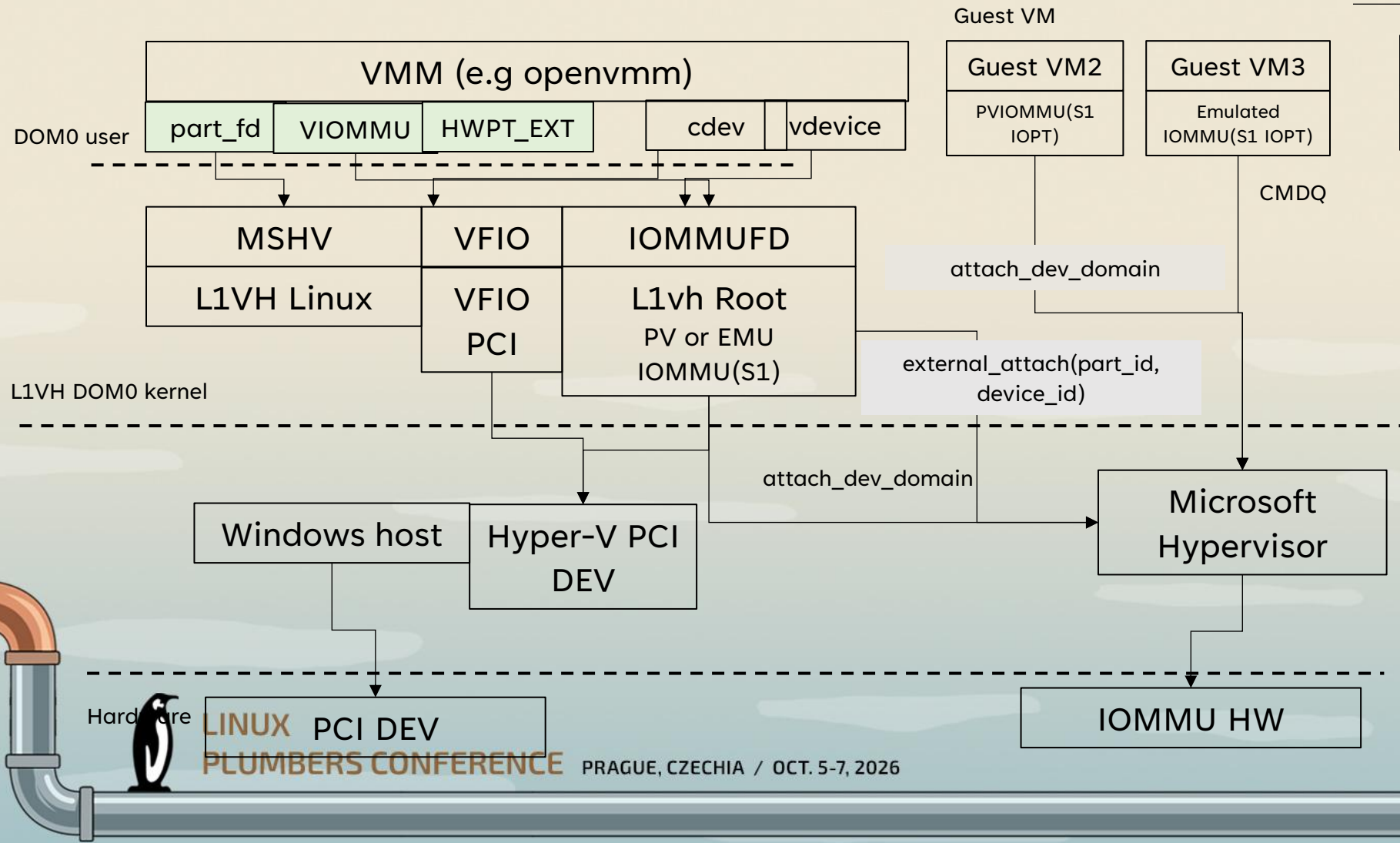


LINUX
PLUMBERS CONFERENCE

PRAGUE, CZECHIA / OCT. 5-7, 2026



DEVICE ASSIGNMENT MODEL: L1VH



Terminology:

- L1vh:** a semi privileged L1 VM that can create and manage guest VMs.
- Emulate IOMMU:** SMMUV3 on GB200
- NO vIOMMU:** DMA to GPA, S1 is PT (pass-through) HWPT
- Partid:** Linked to `vm_fd`, `viommu`, backing S2 IOPT of the `hwpt` is derived from `part_id`
- Device domain** is S2; `l1vh` and `BM dom0` host can have their own S1 and nest on S2 domain allocated from Hyper-V device domains



LINUX PCI DEV
PLUMBERS CONFERENCE

PRAGUE, CZECHIA / OCT. 5-7, 2026