

Intel GVT-g: From Production To Upstream

Zhi Wang zhi.a.wang@intel.com

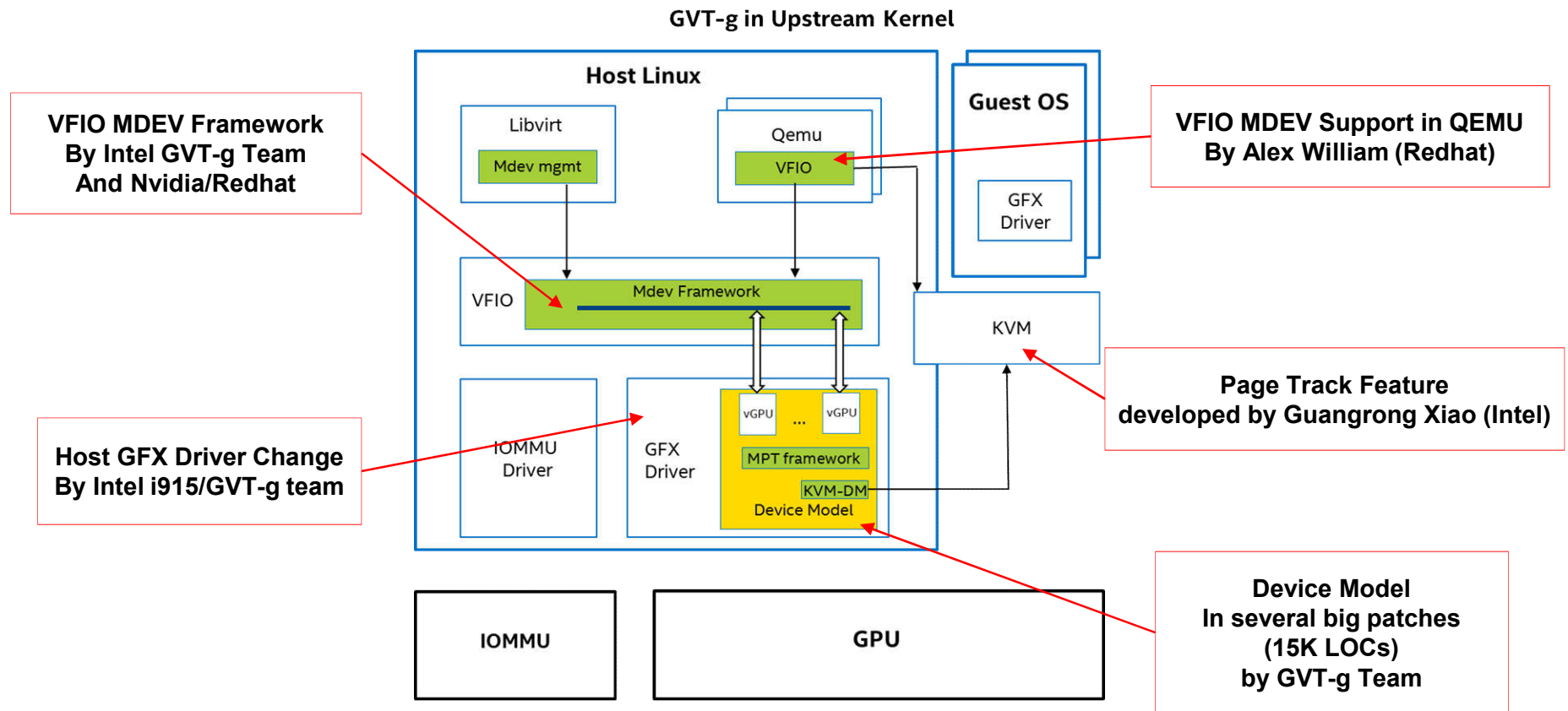


Status

- Official Website: <https://01.org/igvt-g>
- Mailing List: <https://lists.01.org/mailman/listinfo/igvt-g>
- Dev Mailing List: <https://lists.freedesktop.org/mailman/listinfo/intel-gvt-dev>
- Code Repos:
 - Kernel: <https://github.com/01org/gvt-linux/>
 - QEMU: <https://github.com/01org/igvtg-qemu/>
- Quarterly Release Model (The latest release is 2017Q2)
- Citrix: First released with XenClient Enterprise 5.5, now working on XenServer
- Redhat: Released with RHEL 7.4 as a technical preview feature
- Whole project is pushed into upstream in Linux kernel 4.10



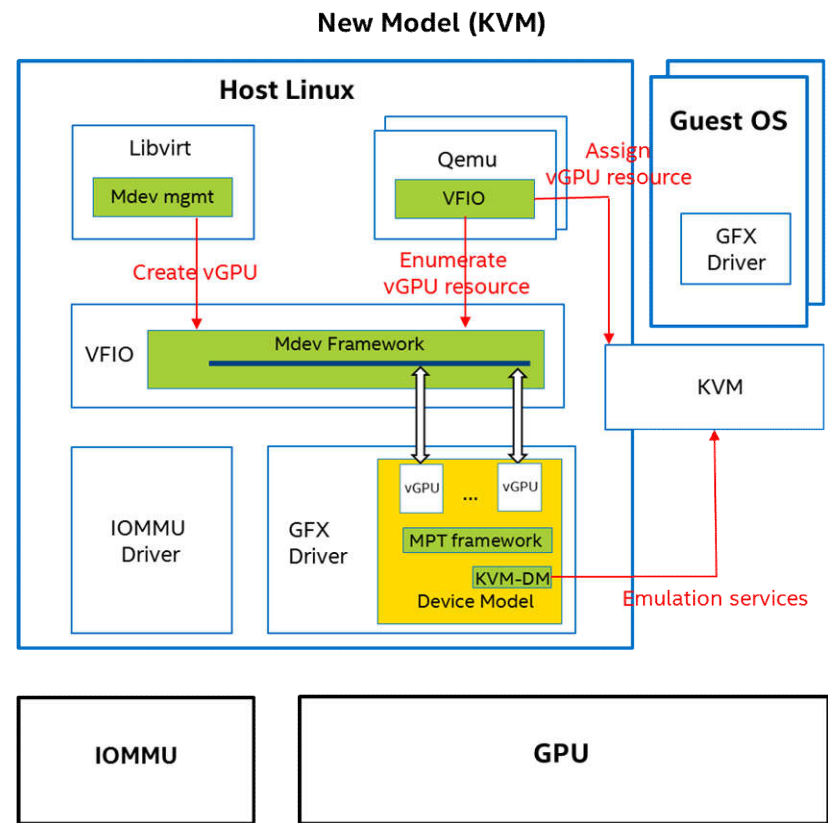
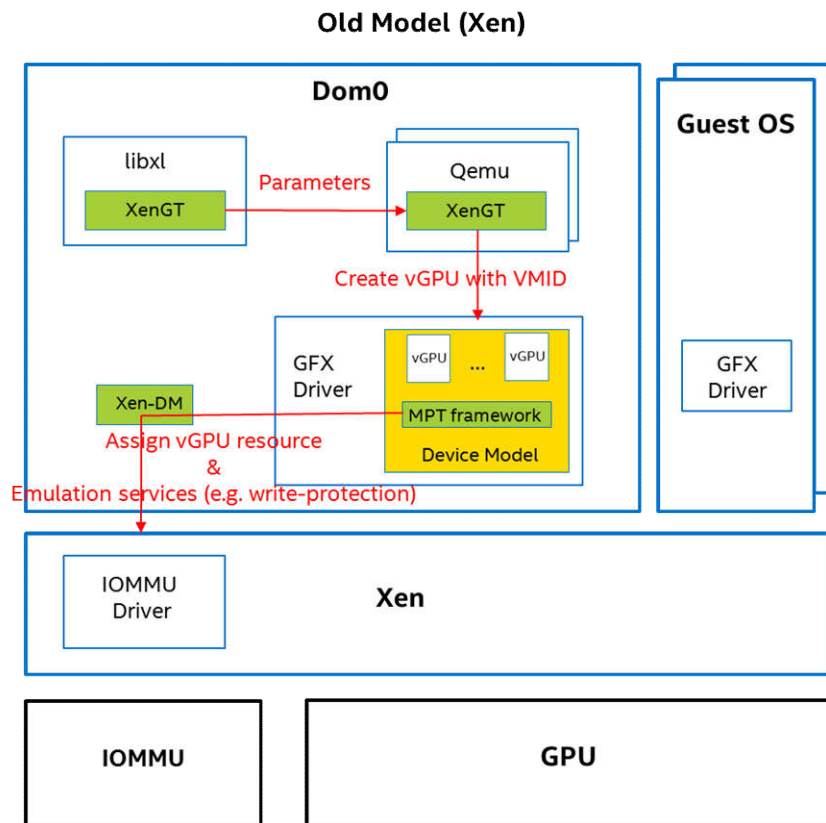
Contributions



Intel GVT-g is supported in Linux Kernel 4.10+ (2016)

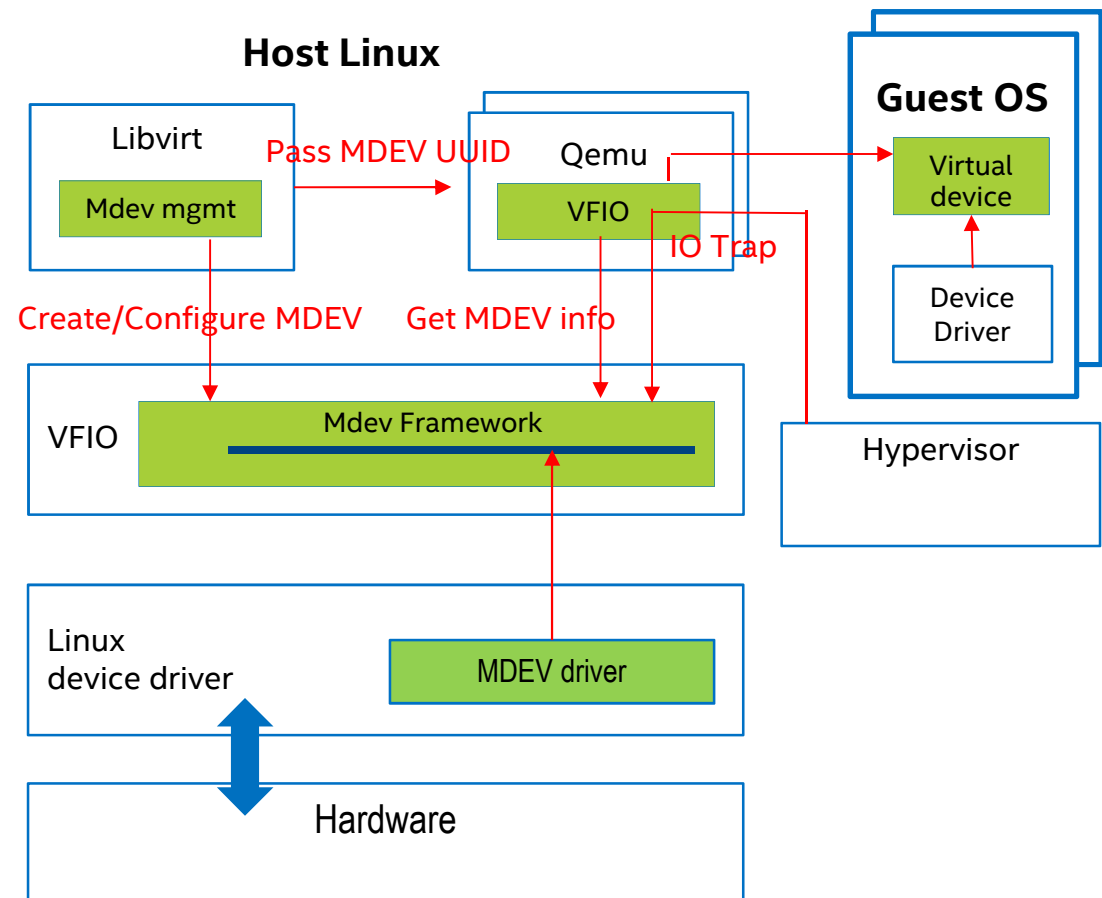


Architecture Changes



Mediated device support in VFIO

- A generic framework for sharing one physical device between VMs
- A MDEV driver only needs to focus on implementing device sharing logics
- Current users in upstream kernel
 - GFX
 - S390 IO driver

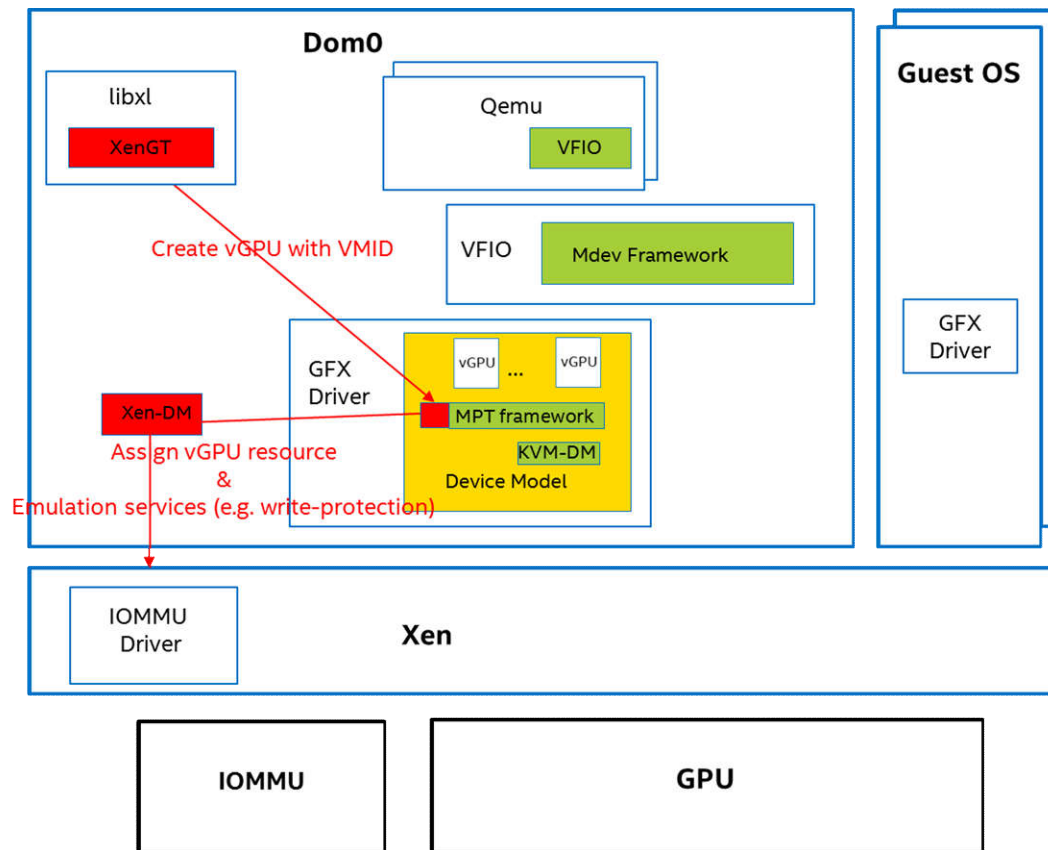


Xen Support Upstream Plan

- Stage 1 – Support both old/new model (bottom line)
 - Avoid XenGT specific changes in Qemu (move to libxl)
 - Intel will upstream two model support in i915 graphics driver
 - Citrix will help upstream libxl changes and xen-dm module
- Stage 2 – Move libxl to use new VFIO mdev mgmt. interface
 - Mdev assignment still stays with Xen specific way thru libxl
- Stage 3 – Fully enable VFIO in Xen
 - Need refactor Qemu and VFIO for Xen support
 - Also benefit passthrough usages



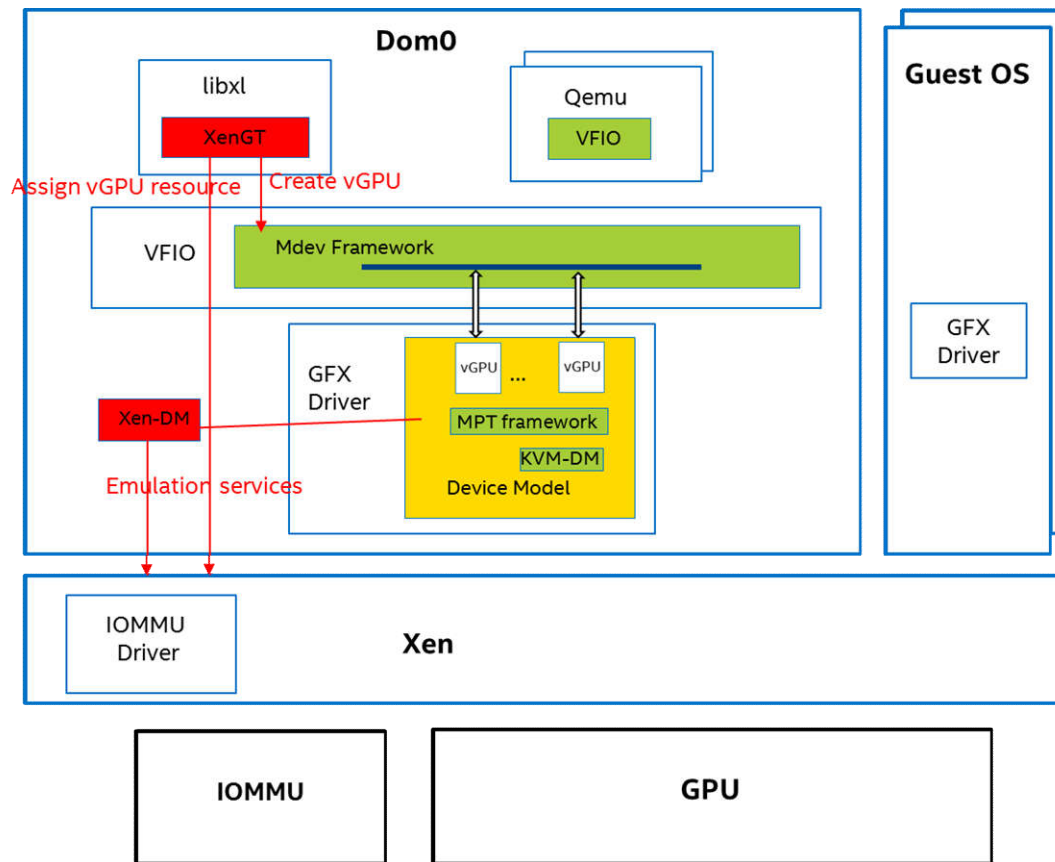
Stage 1 – Support Both Models



- Device model provides mgmt. interfaces for both modes
 - VFIO vs. i915 specific
- Avoid Qemu changes
- Only use libxl for vGPU mgmt
- Xen-DM is a general in-kernel device model framework



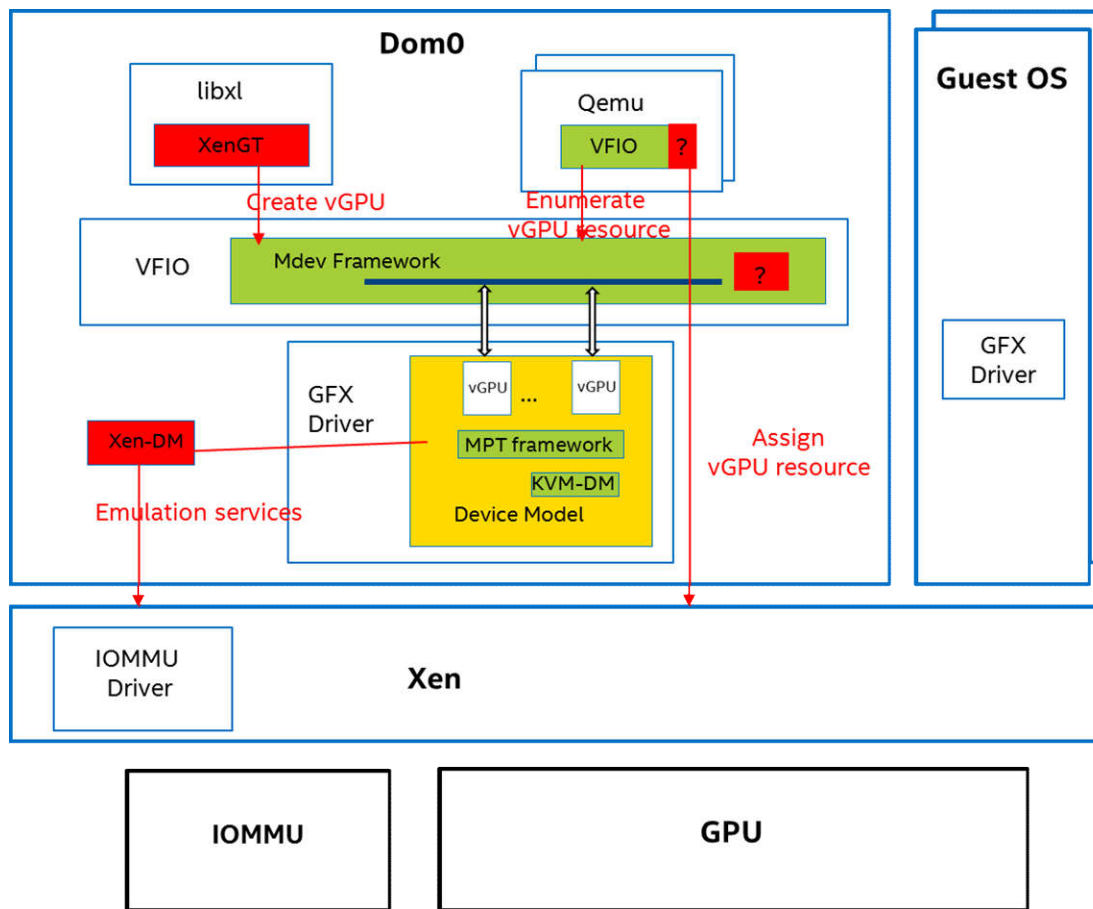
Stage 2 – Use VFIO Mdev Mgmt. Interface



- Use VFIO mdev life-cycle mgmt interface
- Move vGPU resource assignment from kernel to libxl



Stage 3 – Fully Enable VFIO in Xen



- Need to refactor VFIO logic in Qemu/kernel, e.g.
 - Assign vGPU resource to Xen
 - Page accounting vs. foreign page
 - ...
- Long term could benefit passthrough
 - E.g. leverage `iommu_group` concept



PVIOMMU Driver

- PVIOMMU driver is orthogonal here
 - VFIO mdev itself doesn't require IOMMU
 - Required for other purposes
 - Avoid introducing new P2M translation hypercall
 - Avoid pinning all guest memory



Call for action

Share your ideas about Xen on VFIO and also welcome for design discussions in the mailing list ☺

- MDEV Support in Xen
- PCI device passthrough support in Xen based on VFIO



Thanks



TURE INTEL LINUX WIRELESS GUPNP KVM POKY
OP CS YOCTO CONNMAN XEN OFONO LINUX KERNEL
SYNCEVOLUTION SIMPLE FIRMWARE INTERFACE (SFI) ENTERPRISE SECURITY INFRASTRUCTURE



INTEL OPEN SOURCE
TECHNOLOGY CENTER