Generic Buffer Sharing Mechanism for Mediated Devices

Tina Zhang
tina.zhang@intel.com
Agenda

Background

Generic Buffer Sharing in MDEV Framework

Status

Summary
Virtual Function I/O

- Secure, userspace driver framework
- Assigns physical I/O device to VMs
Mediated Device in VFIO

Host Linux

Libvirt
Mdev mgmt

VFIO
Mdev Framework

Linux device driver
MDEV driver

Hardware

Qemu
VFIO

Guest OS

Device Driver

Hypervisor

Pass MDEV UUID
Get MDEV info
I/O Trap
Create/Configure MDEV
A KEY MDEV Usage Requirement in GPU Virtualization Is

To Render Guest Framebuffer In The Host Side
Scenario-1: Remote Virtual Desktop

Desktop#1 ➔ 3D/Media Acceleration ➔ Desktop#2 ➔ ... ➔ Desktop#N

Remote Framebuffer Streaming

Network
Scenario-2: Automotive Virtualization

GPU Mediator Directly Configure VM Framebuffer to Display Panel
Scenario-3: Rich Virtual Client

VM1
framebuffer

VM2
framebuffer

VM3
framebuffer

Guest Framebuffers Composited by Host Compositor
Generic Buffer Sharing in MDEV Framework
Design Philosophy - Generic

• Based on dma-buf subsystem which
  • Generic way provided by kernel to share DMA buffers
  • Already been supported by many device drivers
  • Supported by Linux graphics stacks
  • Used by remote protocol (e.g. SPICE)

• ABIs proposed by different MDEVs in the same category
A Short Note On Dma-buf

The dma-buf
- Represents a sg_table (SGT)

The exporter
- Producer of the dma-buf
- Implements dma_buf_ops
- Exports dma-buf as a file descriptor

The buffer-user
- Imports the dma-buf
- Gets the scatterlist in SGT
- Maps buffer address
Buffer Sharing in MDEV Framework

Importing Buffers

User space

Kernel space

Exporting Buffers

VFIO

Mdev Framework

DMA-BUF subsystem

Linux Driver

Buffer User

Imports Pages

Memory

Mediated Device Driver (Buffer Exporter)

DMA-BUF mgmt

dma-buf
dma-buf
... dma-buf

SGT mgmt

Exported Pages

Exported Pages

Exported Pages

New Component

Existing Component

MDEV buffer

MDEV buffer

MDEV buffer
Buffer Sharing in MDEV Framework

Importing Buffers

Exporting Buffers

User space

Kernel space

VFIO

Mdev Framework

Query Buffer Info

DMA-BUF subsystem

Linux Driver

Buffer User

Imports Pages

Memory

Mediated Device Driver (Buffer Exporter)

DMA-BUF mgmt

dma-buf
dma-buf
dma-buf

SGT mgmt

Exported Pages

Exported Pages

Exported Pages

MDEV buffer

MDEV buffer

MDEV buffer

New Component

Existing Component
Buffer Sharing in MDEV Framework

Importing Buffers

Exporting Buffers

User space

Kernel space

VFIO

Mdev Framework

DMA-BUF subsystem

Linux Driver

Buffer User

Imports Pages

Memory

Mediated Device Driver (Buffer Exporter)

DMA-BUF mgmt

dma-buf
dma-buf

SGT mgmt

Exported Pages

Exported Pages

Exported Pages

MDEV buffer

MDEV buffer

MDEV buffer

Ask for dma-buf fd of the buffer

Return fd of the exposed buffer

New Component

Existing Component
Buffer Sharing in MDEV Framework

Importing Buffers ! Exporting Buffers
Pass fd

User space
Kernel space

DMA-BUF subsystem

VFIO
Mdev Framework

Mediated Device Driver (Buffer Exporter)
DMA-BUF mgmt
dma-buf   dma-buf   ...   dma-buf
SGT mgmt
Exported Pages Exported Pages ... Exported Pages

Linux Driver
Buffer User
Imports Pages

Memory
MDEV buffer MDEV buffer MDEV buffer

New Component
Existing Component
Buffer Sharing in MDEV Framework

Importing Buffers

Exporting Buffers

Access the buffer through fd

VFIO

Mdev Framework

DMA-BUF subsystem

Linux Driver

Buffer User
Imports Pages

Memory

Mediated Device Driver (Buffer Exporter)

DMA-BUF mgmt

dma-buf
dma-buf
...dma-buf

SGT mgmt

Exported Pages

Exported Pages
...Exported Pages

Existing Component

New Component

MDEV buffer
MDEV buffer
MDEV buffer
VFIO_DEVICE_QUERY_GFX_PLANE

```c
struct vfio_device_gfx_plane_info {
    /* in */
    __u32 drm_plane_type; /* type of plane: DRM_PLANE_TYPE_* */
    /* out */
    __u32 width;        /* width of plane */
    __u32 height;       /* height of plane */
    __u32 stride;       /* stride of plane */
    __u32 size;         /* size of plane in bytes, align on page*/
...
    __s32 dmabuf_id;    /* dma-buf id */
};
```
VFIO DEVICE GET GFX DMABUF

struct vfio_device gfx_dmabuf_fd {
    __u32 argsz;
    __u32 flags;
    /* in */
    __u32 dmabuf_id;
    /* out */
    __s32 dmabuf_fd;
};
Sharing vGPU Framebuffer For Remote Desktop

Qemu Remote Protocol Server

VFIO display

VFIO Mdev Framework

Linux Driver

GFX Driver

Mediated GFX Driver

DMA-BUF mgmt

dma-buf dma-buf ... dma-buf

SGT mgmt

Exported Pages Exported Pages Exported Pages

Memory

VM1 framebuffer VM2 framebuffer VM3 framebuffer

Desktop#1 Desktop#2 Desktop#N...

3D/Media Acceleration

Remote Framebuffer Streaming

Network
Sharing vGPU Framebuffer For Direct Display

1. `drmPrimeFDToHandle`: `dma-buf fd -> handle`

2. `drmModeAddFB2`: `handle -> framebuffer`

3. `drmModeSetCrtc`: render the framebuffer to local display panel

```
drmPrimeFDToHandle: drmPrimeFDToHandle(dma-buf fd) -> handle

drmModeAddFB2: drmModeAddFB2(handle) -> framebuffer

drmModeSetCrtc: drmModeSetCrtc(... , framebuffer, handle, crtc) ;
  render the framebuffer to local display panel

```

**Diagram:**

- **Qemu**
- **VFIO**
- **libDRM**
- **Display Switcher**
- **Mediated GFX Driver**
  - DMA-BUF mgmt
  - SGT mgmt
  - Exported Pages
  - Imported Pages
- **Linux Driver**
- **Memory**
- **Physical Display A**
- **Physical Display B**
- **Physical Display C**
- **VM1 framebuffer**
- **VM2 framebuffer**
- **VM3 framebuffer**
Sharing vGPU Framebuffer For Composited Display

eglCreateImageKHR: Dma-buf fd -> image

glBindTexture: Bind image to texture
Implementation Details

DMA-BUF Management
- Implements dma-buf operations
- Creates dma-buf object with an installed fd
- Caches the exposed DMA-BUF object

SGT Management
- Allocates/Releases the SGT
- Gets address of the mediated device buffer

Guest buffer Information Collection
- Gets the info by decoding values in the mediated device registers
Status

PATCH v15:
- Tested with Linux kernel 4.13
- Working with community for upstreaming

Userspace:
- Qemu UI patch-set of “start adding dma-buf support” designed by Gerd Hoffman
- Qemu UI RFC patch-set of “add display support” designed by Gerd Hoffman
Summary

- The design is generic enough for I/O mediators.
  - Based on DMA-BUF subsystem

- The implementation is generic for mediated vGPUs.

- Welcome innovations based on this generic buffer sharing mechanism