virtio-loopback & gpu

As for GPU several versions of the devices are available

- GPU
  - vhost-user-gpu
    - C implementation, well tested and broadly used
  - Virtio-gpu-rutabaga (Rust), cross-platform abstraction for GPU and display virtualization
    - Based on the VIRTIO_GPU_CONTEXT_INIT function in the kernel, simplifies user space
    - maturing rapidly
Vhost-user-gpu picture

- Virtio-loopback-adapter
- Virtio-GPU-support
- Vhost-user-gpu device
  - Virtio-GPU device model
  - SDL2/EGL rendering backend
- GLmark2
- Weston
- contrib/vhost-user-gpu
- Mesa library
- Virtio-gpu
  - /dev/dri/card & render
  - probe
- Virtio-loopback
  - /dev/loopback
- GPU driver (e.g., AMDGPU)

vhost-user-gpu/ QEMU specific components

User-space
Kernel-space
Vhost-user-gpu description

- Vhost-user-gpu requires an external application (QEMU:`contrib/vhost-user-gpu`) to serve the composition requests.
- It uses vhost-user-gpu, an custom extension of vhost-user protocol, to communicate between the vhost-user-gpu and QEMU.
  - It encapsulates requests from allocating memory for graphic-related objects and the emulation of a 'hardware vsync'/frame update.
vhost-user-gpu status

- vhost-user-gpu and virtio-gpu device models development/porting from QEMU to the adapter
  - vhost-user-gpu protocol interactions took place, without the ability to have normalized and contiguous rendered frames
  - Vhost-user-gpu protocol issues – composition side – redraw and allocation of buffers don’t work
- Rendering infrastructure to instantiate and render frames of the virtio-gpu device via the SDL library
  - The SDL foundations were ported from QEMU, and tested with frames stored on the filesystem (not virtio-gpu realtime).
  - To be used in the non sommellier use case of rutabaga
- Under further examination
Rutabaga supports two use cases:

- **QEMU-like**: Instance a new compositor
  - Same as vhost-user-gpu – i.e., guest runs compositor
  - A new compositor is run to support the virtio-gpu application

- **Sommelier**: Use the compositor of the host
Rutabaga supports two use cases:

- **QEMU-like**: Instance a new compositor
  - Same as vhost-user-gpu
  - A new compositor is run to support the virtio-gpu application

- **Sommelier**
  - Use the same compositor of the host (efficiency)
Pictures comparison

Virtio-loopback-adapter
Virtio-GPU-support
Vhost-user-gpu device
Virtio-GPU device model
SDL2/EGL rendering backend

GLmark2
Weston
Mesa library

Virtio-loopback /dev/loopback
Virtio-gpu /dev/dri/card & render

GPU driver (e.g., AMDGPU)

sommenier use-case

GLmark2
Wayland host compositor e.g., Weston
Wayland socket

Virtio-gpu-rutabaga

crosvm/rutabaga library (gfx_ffi)

Virtio-GPU-support

Sommelier invocation

User-space

User-space

kernel-space

Kernel-space

probe

probe

Virtual Open Systems
Virtio-gpu-rutabaga

- Rutabaga (rust) is part of crosvm, sommelier (C++) is part of chromium os
- Can use system resources more efficiently, because we are not obliged to run additional weston compositors
- In the sommelier deployment, applications are launched by the use of the launcher application `sommelier`
  - Sommelier is a wayland compositor that performs rendering operations
- Simplify the architecture removing vhost-user-gpu protocol and vhost-user-gpu
  - Better maintenance
  - Easier porting
- No need to modify existing virtio-gpu applications, the same used with vhost-user-gpu can be executed – in any deployment type
Virtio-gpu-rutabaga status

- All the software components have been ported or implemented
- Both use cases will be eventually supported, starting from the sommelier one
- At the moment we are verifying that all the connected parts have been wired correctly.
- Functional tests and benchmarks will follow
Use case discussion

Use case with

- Two applications (or containers) want to use virtio-GPU
- The system is using virtio-loopback
Use case discussion - vhost-user-gpu

- Two applications
- 2+ weston compositors
- Two virtio-gpu
- Two vhost-user-gpu
Use case discussion - rutabaga

- Two applications
- Two—one weston compositors
- Two One virtio-gpu
- Two vhost-user-gpu
Use case-based comparison

- **Vhost-user-gpu**
  - At least two weston instances (one for the host, one for the application that you want to render)

- **Rutabaga**
  - Efficient use of weston instances (lightweight compositor integrated in sommelier)
  - Supports both QEMU-like and sommelier use case
  - No need to modify existing applications
  - Less components, easier to maintain
Conclusions

- vhost-user-gpu works very well in use cases where a full system is virtualized (applications + weston, as in the first two use cases below)
- Rutabaga gives in addition the possibility to share a single (host) weston instance
  - This solution leads to a simpler and more efficient solution, with no compromises at functional level
Proposed next steps

➢ Get a first rutabaga demo
   ➢ Some test will be done on the virtio-loopback side to see if the rendering node is behaving as expected (simple DRM tests)
   ➢ The sommelier use case will be tested first as it involves less moving parts
contact@virtualopensystems.com
Web: virtualopensystems.com
Demos: virtualopensystems.com/en/solutions/demos/
Guides: virtualopensystems.com/en/solutions/guides/
Research projects: virtualopensystems.com/en/research/innovation-projects/