



AGL Native VIRTIO (Phase 2) Status and next steps



Michele Paolino
2023-01-17

m.paolino@virtualopensystems.com
www.virtualopensystems.com



Virtio-loopback phase 2 objectives

The list of the project task are:

- Task 1: virtio-loopback support for virtio-GPU
- Task 2: virtio-loopback support for virtio-sound
- Task 3: virtio-loopback support for virtio-CAN
- Task 4: virtio-loopback support for virtio-GPIO and console
- Task 5: Apply Non-HV VirtIO to Cloud

For tasks 1-4, the target is to use existing vhost-user device backends (GPU, sound, GPIO) or implement them when needed (CAN and console).



Task 1: GPU - activity description

- Add adapter support for vhost-user-gpu
 - Use QEMU's implementation 'hw/display/virtio-gpu*'
 - Isolate & extract dependencies from QEMU
 - Re-write systems which are not ported from QEMU
 - e.g., QEMU character device abstraction
 - Used to operate socket communication
 - Timer classes
 - Object oriented device initializations
 - QEMU Type system & parent/child relationships



Task 1: GPU - activity description

Windowing system

- Adding rendering capability to the adapter (same as QEMU)
 - Selected the SDL/EGL backend to set the scene based on how QEMU does it (discarded GTK)
 - Extracted the dependent functionality from QEMU
 - DisplayChangeListener API and SDL API
 - Integration of SDL window initialization on adapter



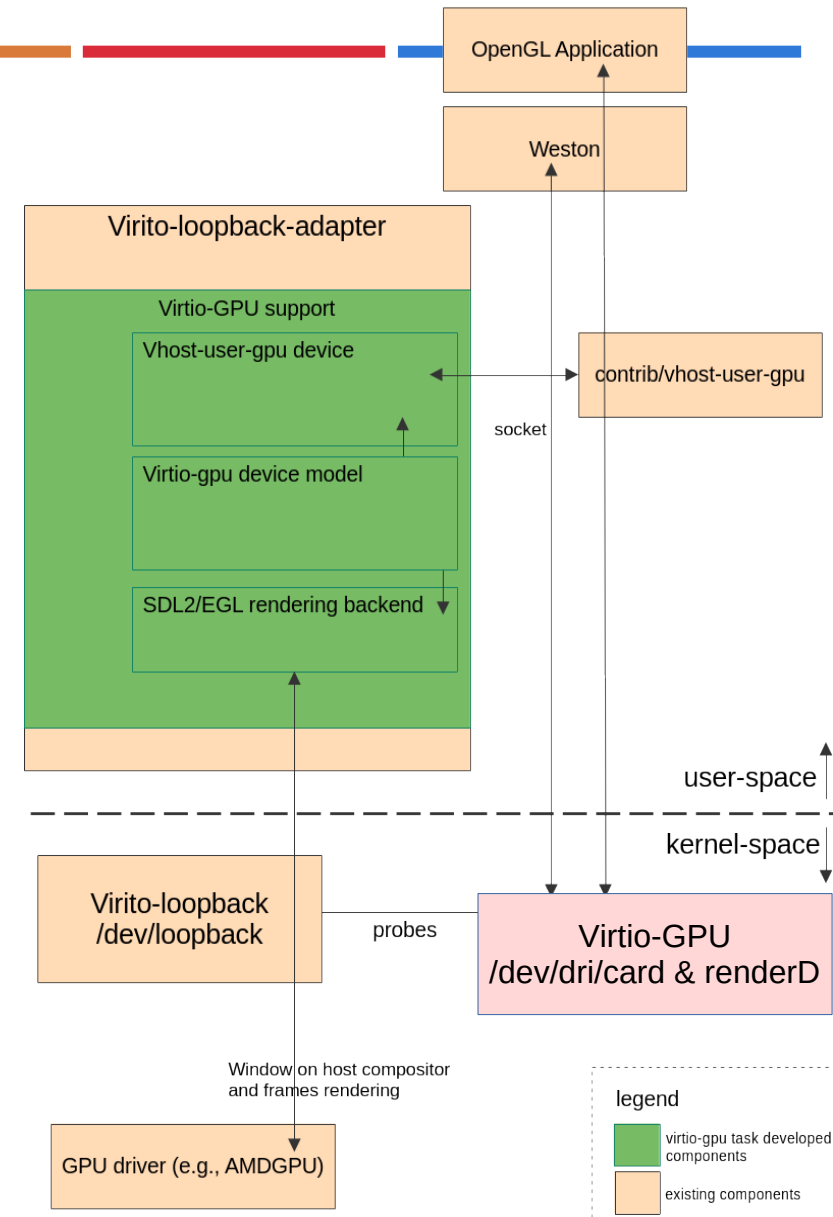
Task 1: GPU - architecture overview

➤ Adapter support for the virtio-gpu split in 3 parts

- Vhost-user-gpu interface
- Virtio-gpu device model
- SDL rendering
 - Uses the real-GPU to render on the host compositor

➤ Sequence of actions

- Virtio-loopback probes the virtio-gpu driver upon adapter request
- Initial SDL window creation
- Weston & application targets the virtio-gpu exposed DRM device files
- Frames are rendered by the adapter upon 'contrib/vhost-user-gpu' requests





Task 1: status overview

- Virtio-loopback-adapter integration status
 - Vhost-user-gpu interface [initial version done]
 - QEMU structures/functionality used by virtio-gpu
 - Virtio-gpu device model [WIP]
 - SDL rendering [initial version done]
 - Frames updates based on vhost-user-gpu requests
- Yocto integration
 - Recipe creation [WIP]
 - fixing libpixman/virgilrender dependency issues



Task 1: status

The screenshot displays a Linux desktop environment with several windows open. The top window is a terminal window showing system logs for the virtio-loopback-adapter. The logs indicate that the adapter is running and connected to the /tmp/vgpu.sock socket. The logs also show the adapter's configuration and the results of various system calls. The bottom-left window is a terminal window showing the output of the weston command, which is used to start the Weston compositor. The bottom-right window is a glmark2 benchmark window showing a score of 5 and OpenGL information. The middle window is a QEMU window showing a 3D rendering of a horse. The top-right window is a Weston compositor window showing the desktop environment.

```
root@clxps:~/home/x/virtio-gpu/virtio_loopback_adapter
vhost-user-loopback: VHOST_USER_GET_QUEUE_NUM
10923adapter vhost_user_loopback.c:155 vu_message_write Called with vhost-user-message 'VHOST_USER_GET_QUEUE_NUM'
10923adapter vhost_user_loopback.c:217 vu_message_read Called with vhost-user-message 'VHOST_USER_GET_QUEUE_NUM'
vhost-user-loopback: Get value: 0x2
10923adapter vhost_user_loopback.c:1556 vhost_user_backend_init Device max_queues 2 num_queues 0
vhost-user-loopback: VHOST_USER_GET_MAX_MEM_SLOTS
10923adapter vhost_user_loopback.c:155 vu_message_write Called with vhost-user-message 'VHOST_USER_SET_STATUS'
10923adapter vhost_user_loopback.c:217 vu_message_read Called with vhost-user-message 'VHOST_USER_SET_STATUS'
vhost-user-loopback: Get value: 0x20
10923adapter vhost_user_loopback.c:1605 vhost_user_backend_init Setup slave channel
10923adapter vhost_user_loopback.c:484 vhost_setup_slave_channel features HAS_VHOST_USER_PROTOCOL_F_SLAVE_REQ
10923adapter vhost_user_loopback.c:155 vu_message_write Called with vhost-user-message 'VHOST_USER_SET_SLAVE_REQ_FD'
vhost-user-loopback: Reply is done!
10923adapter vhost_user_loopback.c:1638 vhost_dev_init Set owner
10923adapter vhost_user_loopback.c:155 vu_message_write Called with vhost-user-message 'VHOST_USER_SET_OWNER'
10923adapter vhost_user_loopback.c:1644 vhost_dev_init vhost user get features
vhost-user-loopback: VHOST_USER_GET_FEATURES
10923adapter vhost_user_loopback.c:155 vu_message_write Called with vhost-user-message 'VHOST_USER_GET_FEATURES'
10923adapter vhost_user_loopback.c:217 vu_message_read Called with vhost-user-message 'VHOST_USER_GET_FEATURES'
vhost-user-loopback: Get value: 0x175000001
vhost-user-loopback: Print vhost_dev_init->features: 0x175000001
vhost-user-loopback: Initializing 2 vqs vhdev->nvqs
vhost-user-loopback: Call vhost_user_set_vring_call()
10923adapter vhost_user_loopback.c:155 vu_message_write Called with vhost-user-message 'VHOST_USER_SET_VRING_CALL'
vhost-user-loopback: Call vhost_user_set_vring_call()
10923adapter vhost_user_loopback.c:185 vu_message_write Called with vhost-user-message 'VHOST_USER_SET_VRING_CALL'
vhost-user-loopback: vhdev->backend_features 0x40000000
vhost-user-loopback: vhdev->features 0x175000001
10923adapter vhost_user_gpu.c:995 vhost_user_gpu_realize vhost_user_gpu_realize DONE

root@clxps:~/home/x
west@clxps/home/x # G_MESSAGES_DEBUG=all /home/x/virtio-gpu/qemu/build/contrib/gpu/vhost-user-gpu --virl --socket-path=/tmp/vgpu.sock
*** MESA_GLSL_CACHE_DISABLE is deprecated; use MESA_SHADER_CACHE_DISABLE instead
*** MESA_GLSL_CACHE_DISABLE is deprecated; use MESA_SHADER_CACHE_DISABLE instead

root@clxps:~/home/x
weston --backend=drm-backend.so --drm-device=card0
Date: 2024-01-05 EET
[15:17:14.606] weston 6.0.1
https://wayland.freedesktop.org
Bug reports to: https://gitlab.freedesktop.org/wayland/weston/iss
Build: unknown (not built from git or tarball)
[15:17:14.606] Command line: weston --backend=drm-backend.so --drm-device=card0
[15:17:14.606] OS: Linux, 6.1.2-1222.mative, #1 SMP Sat Dec 31 05:00:12 PST 2022
[15:17:14.606] warning: MDC_RUNTIME_DIR "/run/user/1000" is not configured correctly. Unix access mode must be 0700 (current mode is 700), and must be owned by the user (current owner is UID 1000). Refer to your distribution on how to get it, or http://www.freedesktop.org/wiki/Specifications/basedir-spec on how to implement it.
[15:17:14.606] Starting with no config file.

glmark2 2023.01
=====
OpenGL Information
GL_VENDOR: Mesa
GL_RENDERER: llvmpipe (LLVM 16.0.6, 256 bits)
GL_VERSION: 4.5 (Compatibility Profile) Mesa 23.2.1-arch1.2
Surface Config: buf=32 r=8 g=8 b=8 a=8 depth=32 stencil=0 samples=0
Surface Size: 640x480 windowed
=====
** GLX does not support GLX_EXT_swap_control or GLX_MESA_swap_control!
** Failed to set swap interval. Results may be bounded above by refresh rate.
[build] use-vbo=false[]
```

1) virtio-loopback-adapter running and connected to the '/tmp/vgpu.sock' socket provided by 'contrib/vhost-user-gpu' application



Task 1: status

```
root@clxps:~/home/x/virtio-gpu/virtio_loopback_adapter
vhost-user-loopback: VHOST_USER_GET_QUEUE_NUM
10923adapter vhost_user_loopback.c:155 vu_message_write Called with vhost-user-message 'VHOST_USER_GET_QUEUE_NUM'
10923adapter vhost_user_loopback.c:217 vu_message_read Called with vhost-user-message 'VHOST_USER_GET_QUEUE_NUM'
vhost-user-loopback: Get value: 0x2
10923adapter vhost_user_loopback.c:1556 vhost_user_backend_init Device max_queues 2 num_queues 0
vhost-user-loopback: VHOST_USER_GET_MAX_MEM_SLOTS
10923adapter vhost_user_loopback.c:155 vu_message_write Called with vhost-user-message 'VHOST_USER_SET_STATUS'
10923adapter vhost_user_loopback.c:217 vu_message_read Called with vhost-user-message 'VHOST_USER_SET_STATUS'
vhost-user-loopback: Get value: 0x20
10923adapter vhost_user_loopback.c:1605 vhost_user_backend_init Setup slave channel
10923adapter vhost_user_loopback.c:484 vhost_setup_slave_channel features HAS VHOST_USER_PROTOCOL_F_SLAVE_REQ
10923adapter vhost_user_loopback.c:155 vu_message_write Called with vhost-user-message 'VHOST_USER_SET_SLAVE_REQ_FD'
vhost-user-loopback: Reply is done!
10923adapter vhost_user_loopback.c:1638 vhost_dev_init Set owner
10923adapter vhost_user_loopback.c:155 vu_message_write Called with vhost-user-message 'VHOST_USER_SET_OWNER'
10923adapter vhost_user_loopback.c:1644 vhost_dev_init vhost user get features
vhost-user-loopback: VHOST_USER_GET_FEATURES
10923adapter vhost_user_loopback.c:155 vu_message_write Called with vhost-user-message 'VHOST_USER_GET_FEATURES'
10923adapter vhost_user_loopback.c:217 vu_message_read Called with vhost-user-message 'VHOST_USER_GET_FEATURES'
vhost-user-loopback: Get value: 0x175000001
vhost-user-loopback: Print vhost_dev_init->features: 0x175000001
vhost-user-loopback: Initializing 2 vqs vhdev->nvqs
vhost-user-loopback: Call vhost_user_set_vring_call()
10923adapter vhost_user_loopback.c:155 vu_message_write Called with vhost-user-message 'VHOST_USER_SET_VRING_CALL'
vhost-user-loopback: Call vhost_user_set_vring_call()
10923adapter vhost_user_loopback.c:155 vu_message_write Called with vhost-user-message 'VHOST_USER_SET_VRING_CALL'
vhost-user-loopback: vhdev->backend_features 0x40000000
vhost-user-loopback: vhdev->features 0x175000001
10923adapter vhost_user_gpu.c:995 vhost_user_gpu_realize vhost_user_gpu_realize DONE

root@clxps:~/home/x
west@clxps/home/x # G_MESSAGES_DEBUG=all /home/x/virtio-gpu/qemu/build/contrib/gpu/vhost-user-gpu --virl --socket-path=/tmp/vgpu.sock
*** MESA_GLSL_CACHE_DISABLE is deprecated; use MESA_SHADER_CACHE_DISABLE instead

root@clxps:~/home/x
west@clxps/home/x # weston --backend=drm-backend.so --drm-device=card0
Date: 2024-01-05 EET
[15:17:14.606] weston 6.0.1
https://wayland.freedesktop.org
Bug reports to: https://gitlab.freedesktop.org/wayland/weston/iss
Build: unknown (not built from git or tarball)
[15:17:14.606] Command line: weston --backend=drm-backend.so --drm-device=card0
[15:17:14.606] OS: Linux, 6.1.2-1222.mative, #1 SMP Sat Dec 31 05:00:12 PST 2022
[15:17:14.606] warning: MDC_RUNTIME_DIR="/run/user/1000" is not configured correctly. Unix access mode must be 0700 (current mode is 700), and must be owned by the user (current owner is UID 1000). Refer to your distribution on how to get it, or http://www.freedesktop.org/wiki/Specifications/basedir-spec on how to implement it.
[15:17:14.606] Starting with no config file.

gltmark2 2023.01
=====
OpenGL Information
GL_VENDOR: Mesa
GL_RENDERER: llvmpipe (LLVM 16.0.6, 256 bits)
GL_VERSION: 4.5 (Compatibility Profile) Mesa 23.2.1-arch1.2
Surface Config: buf=32 r=8 g=8 a=8 depth=32 stencil=0 samples=0
Surface Size: 640x480 windowed
=====
** GLX does not support GLX_EXT_swap_control or GLX_MESA_swap_control!
** Failed to set swap interval. Results may be bounded above by refresh rate.
[build] use-vbo=false[]
```

2) weston compositor started on the DRM device probed by the virtio-loopback-adapter kernel driver via the adapter

- WIP: The application does not recognize the probed driver by virtio-loopback and fallbacks to the default gpu node



Task 1: status

```
root@clxps:~/home/x/virtio-gpu/virtio_loopback_adapter
vhost-user-loopback: VHOST_USER_GET_QUEUE_NUM
10923adapter vhost_user_loopback.c:155 vu_message_write Called with vhost-user-message 'VHOST_USER_GET_QUEUE_NUM'
10923adapter vhost_user_loopback.c:217 vu_message_read Called with vhost-user-message 'VHOST_USER_GET_QUEUE_NUM'
vhost-user-loopback: Get value: 0x2
10923adapter vhost_user_loopback.c:1556 vhost_user_backend_init Device max_queues 2 num_queues 0
vhost-user-loopback: VHOST_USER_GET_MAX_MEM_SLOTS
10923adapter vhost_user_loopback.c:155 vu_message_write Called with vhost-user-message 'VHOST_USER_SET_STATUS'
10923adapter vhost_user_loopback.c:217 vu_message_read Called with vhost-user-message 'VHOST_USER_SET_STATUS'
vhost-user-loopback: Get value: 0x20
10923adapter vhost_user_loopback.c:1605 vhost_user_backend_init Setup slave channel
10923adapter vhost_user_loopback.c:484 vhost_setup_slave_channel features HAS_VHOST_USER_PROTOCOL_F_SLAVE_REQ
10923adapter vhost_user_loopback.c:155 vu_message_write Called with vhost-user-message 'VHOST_USER_SET_SLAVE_REQ_FD'
vhost-user-loopback: Reply is done!
10923adapter vhost_user_loopback.c:1638 vhost_dev_init Set owner
10923adapter vhost_user_loopback.c:155 vu_message_write Called with vhost-user-message 'VHOST_USER_SET_OWNER'
10923adapter vhost_user_loopback.c:1644 vhost_dev_init vhost user get features
vhost-user-loopback: VHOST_USER_GET_FEATURES
10923adapter vhost_user_loopback.c:155 vu_message_write Called with vhost-user-message 'VHOST_USER_GET_FEATURES'
10923adapter vhost_user_loopback.c:217 vu_message_read Called with vhost-user-message 'VHOST_USER_GET_FEATURES'
vhost-user-loopback: Get value: 0x175000001
vhost-user-loopback: Print vhost_dev_init->features: 0x175000001
vhost-user-loopback: Initializing 2 vqs vhdev->nvqs
vhost-user-loopback: Call vhost_user_set_vring_call()
10923adapter vhost_user_loopback.c:155 vu_message_write Called with vhost-user-message 'VHOST_USER_SET_VRING_CALL'
vhost-user-loopback: Call vhost_user_set_vring_call()
10923adapter vhost_user_loopback.c:185 vu_message_write Called with vhost-user-message 'VHOST_USER_SET_VRING_CALL'
vhost-user-loopback: vhdev->backend_features 0x40000000
vhost-user-loopback: vhdev->features 0x175000001
10923adapter vhost_user_gpu.c:995 vhost_user_gpu_realize vhost_user_gpu_realize DONE

root@clxps:~/home/x
root@clxps:~/home/x # G_MESSAGES_DEBUG=all /home/x/virtio-gpu/qemu/build/contrib/gpu/vhost-user-gpu --virl -socket-path=/tmp/vgpu.sock
*** MESA_GLSL_CACHE_DISABLE is deprecated; use MESA_SHADER_CACHE_DISABLE instead

root@clxps:~/home/x
root@clxps:~/home/x # weston --backend=drm-backend.so --drm-device=card0
Date: 2024-01-05 EET
[15:17:14.606] weston 6.0.1
https://wayland.freedesktop.org
Bug reports to: https://gitlab.freedesktop.org/wayland/weston/iss
Build: unknown (not built from git or tarball)
[15:17:14.606] Command line: weston --backend=drm-backend.so --drm-device=card0
[15:17:14.606] OS: Linux, 6.1.2-1222.mative, #1 SMP Sat Dec 31 05:00:12 PST 2022
[15:17:14.606] warning: MDC_RUNTIME_DIR "/run/user/1000" is not configured correctly. Unix access mode must be 0700 (current mode is 700), and must be owned by the user (current owner is UID 1000). Refer to your distribution on how to get it, or http://www.freedesktop.org/wiki/Specifications/basedir-spec on how to implement it.
[15:17:14.606] Starting with no config file.

x@archvm:~
glmark2 2023.01
=====
OpenGL Information
GL_VENDOR: Mesa
GL_RENDERER: llvmpipe (LLVM 16.0.6, 256 bits)
GL_VERSION: 4.5 (Compatibility Profile) Mesa 23.2.1-arch1.2
Surface Config: buf=32 r=8 g=8 b=8 a=8 depth=32 stencil=0 samples=0
Surface Size: 640x480 windowed
=====
** GLX does not support GLX_EXT_swap_control or GLX_MESA_swap_control!
** Failed to set swap interval. Results may be bounded above by refresh rate.
[build] use-vbo=false[]
```

3) glmark2 application startup. The command line includes the `DRI_PRIME=1` to indicate the DRM rendering node associated with this workload

- WIP: The application does not recognize the proposed driver by virtio-loopback and fallbacks to the default gpu node



Task 1: status

1

2

3

4

5

3) Rendering part of the virtio-loopback-adapter rendering the frames coming from the 'contrib/vhost-user-gpu' application



Task 1: conclusions and next steps

- Very challenging task (QEMU side only is ~4500 lines of code)
 - Complexity comes from the different layers in play: QEMU device abstraction, QEMU type system, vhost-user protocol, SDL API, etc.
- We have a first version of the QEMU vhost-user-gpu support in the adapter
 - Able to register a new virtio-gpu device via vhost-user-gpu protocol (communication between adapter and kernel is OK)
 - Able to start rendering window (SDL initialization works)

Next Steps:

- Fix requests from virtio-GPU kernel model to the adapter GPU device model (to fix differences between QEMU device models and adapter)
 - Trial error process: the complexity of QEMU abstraction makes things hard to debug
- We remain committed and we keep working



Task 2: sound activity description

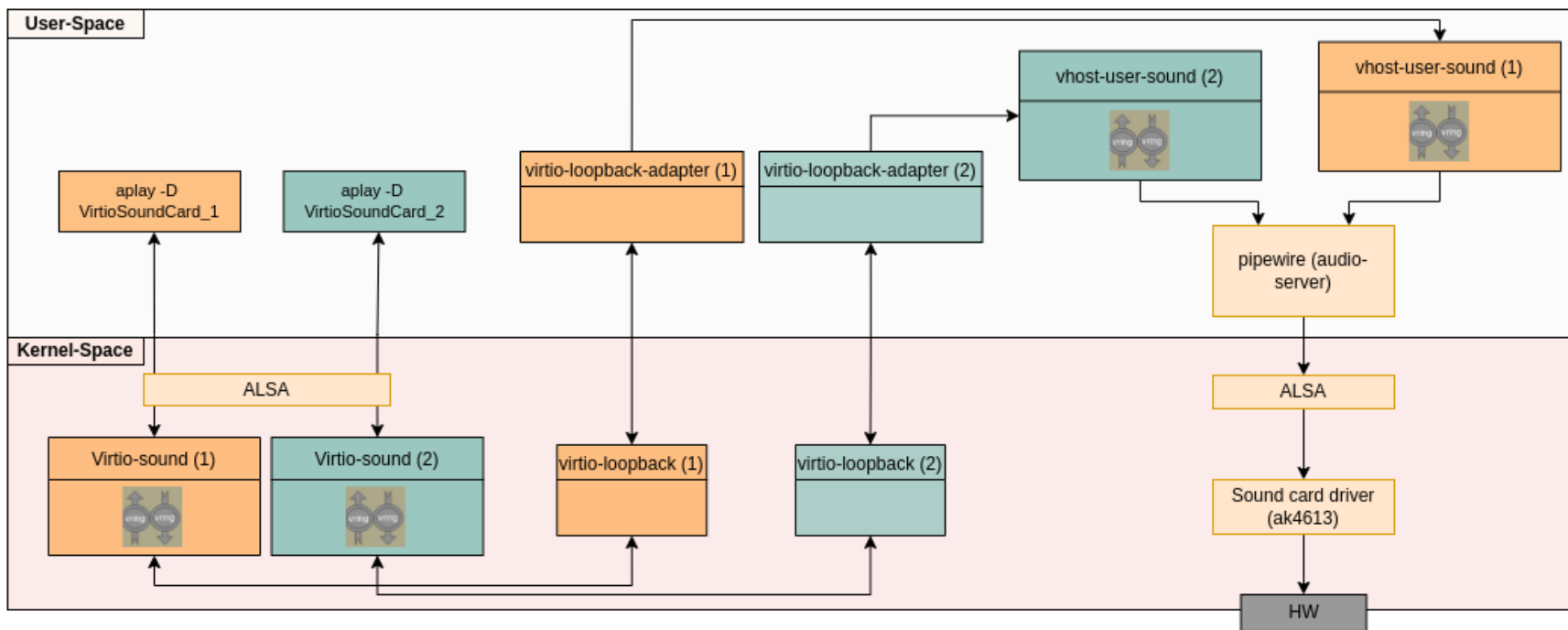
- Work is based on a rust-vmm device under active development
 - No crates.io availability yet
 - Issues with cross-compilation
- Vhost-user-sound adapter support added
- Tested with AGL reference hardware
- Create multi-sound devices support
 - [Additional task activity added during the project]



Task 2: demo on reference hardware

The demonstration (AGL ALS Tokyo) shows **two completely separated sound stacks** running on top of virtio-loopback architecture:

- Two applications (“aplay”) route their audio via the
- Two virtio-loopback drivers and vhost-user-sound devices



Both audio streams converge in “pipewire” audio server which can manage/prioritize the audio streams!



Task 2: sound status

- Development and functional testing completed



Task 2: next steps

- Create yocto recipe
 - The lack of crates.io support for vhost-user-sound obliges us to a “manual” approach
 - Issues with cross compilation (rust AGL yocto support)
 - Current recipe works with qemux86_64 target but doesn't with arm based targets



Task 3: CAN activity description

- Vhost-user-can device not available at the beginning of the task. We need to create it
- Existing virtio-can driver RFC (SocketCAN) based on virtio-can RFC code from OpenSynergy (<https://lwn.net/Articles/934187/>) can be reused
- Enable vhost-user device support in the virtio adapter
- Yocto integration
- Push contributions back to rust-vm



Task 3: CAN status

- Task 3 – virtio-loopback support for virtio-CAN
 - Created vhost-user version of virtio-can driver that uses Linux kernel pre-existing device driver (SocketCAN)
 - Vhost-user-can Gerrit:
<https://gerrit.automotivelinux.org/gerrit/c/AGL/meta-agl-devel/+29407>
 - Enabled vhost-user device support in the virtio adapter
 - Gerrit link: <https://gerrit.automotivelinux.org/gerrit/c/src/virtio/virtio-loopback-adapter/+29493>
 - Proposed the newly created device to rust-vmm community
 - Pull request on rust-vmm: <https://github.com/rust-vmm/vhost-device/pull/602>



Task 3: next steps

- Support existing pull request addressing community requests
 - Pull request on rust-vmm:
<https://github.com/rust-vmm/vhost-device/pull/602>



Task 4: activity description

GPIO & Console

- GPIO
 - Vhost-user-device available
 - Add support for the adapter and prepare yocto recipe
- Console
 - No vhost-user-device available (same as CAN)
 - Add support for the adapter and prepare yocto recipe
 - Push contributions back to rust-vmm



Task 4: status GPIO & Console

➤ GPIO

- Enabled vhost-user device support in the virtio adapter
 - **Gerrit link:** <https://gerrit.automotivelinux.org/gerrit/c/AGL/meta-agl-devel/+29390>
- **Added vhost-user-gpio in AGL:**
<https://gerrit.automotivelinux.org/gerrit/c/AGL/meta-agl-devel/+29407>

➤ Console

- Create vhost-user version of the driver (virtio-console only) that uses Linux kernel pre-existing device driver
 - <https://gerrit.automotivelinux.org/gerrit/c/AGL/meta-agl-devel/+29545>
- Enable vhost-user device support in the virtio adapter
 - <https://gerrit.automotivelinux.org/gerrit/c/src/virtio/virtio-loopback-adapter/+29539>
- Pull request on rust-vmm done
 - <https://github.com/rust-vmm/vhost-device/pull/601>



Task 4: next steps

- Support existing pull request addressing community requests
 - Pull request on rust-vmm:
<https://github.com/rust-vmm/vhost-device/pull/601>



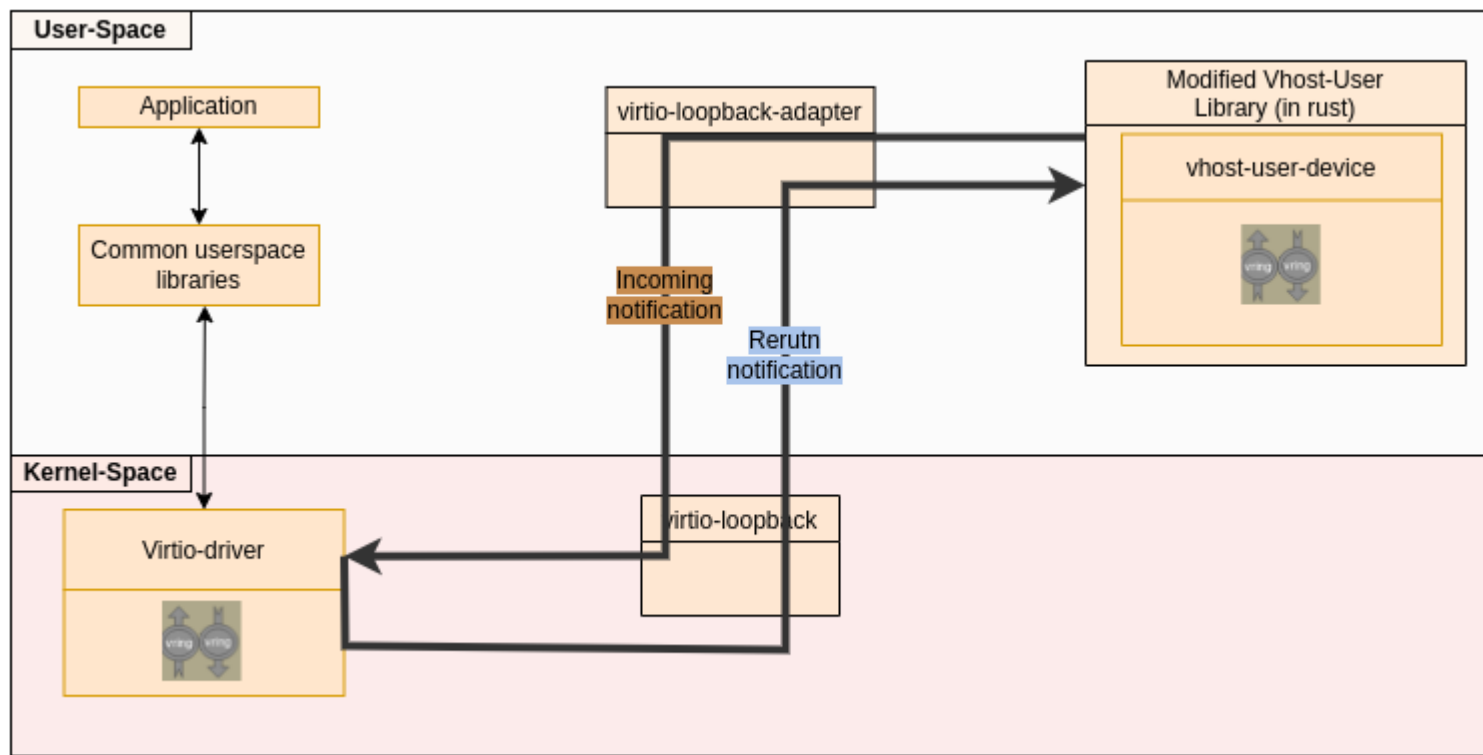
Task 1-4 : virtio-loopback performance

- Measure virtio-loopback performance infrastructure targeting:
 - Interrupt notification
 - Throughput
- Virtio-loopback will then be compared with QEMU/KVM vhost-user performance in the same conditions
- GPU, being a particularly interesting target performance wise, will be tested with specific benchmark
- Status:
 - Interrupt notification benchmark is almost final
 - for the throughput benchmark we will use blk



Task 1-4 : virtio-loopback performance

- Very preliminary first version of latency measurement
 - Average: 0.901 ms
 - Standard Deviation: 0.157 ms





Task 5: activity description and next steps

- All the virtio-loopback development described so far should be executable also on the Amazon AWS cloud
- To do this, QEMU arm64 and x86_64 were added as a build target
- Next step
 - Ensure that all the components keep building for all the targets (QEMU arm and x86_64 and reference hardware)



Wrap up: Work in progress (WIP)

The list of the project task are:

- Task 1: virtio-loopback support for virtio-GPU
 - Trial/error QEMU integration
 - Yocto recipes
- Task 2: virtio-loopback support for virtio-sound
 - Yocto recipes
- Task 5: aws support
 - Run virtio-loopback in AWS
- [Additional] virtio-loopback tests for AGL CI
 - not started



Wrap up: Completed

The list of the project task are:

- Task 3 and 4 (CAN, GPIO, console and Cloud)
 - Completed, code maintenance up to the end of the project
- [Additional] virtio-loopback infrastructure improvement and bug fixes
 - Create vhost-user compatibility layer (no changes required in vhost-user protocol)
 - Add Multi-queue support
- [Additional] multi device support demonstration



Questions



?





contact@virtualopensystems.com

Web: virtualopensystems.com

Products: <http://www.virtualopensystems.com/en/products/>

Demos: virtualopensystems.com/en/solutions/demos/

Guides: virtualopensystems.com/en/solutions/guides/

Research projects: virtualopensystems.com/en/research/innovation-projects/