



云原生技术开源开放日

最新云桌面开源成果分享： **Virtio GPU**视频加速框架解析

江 锋

openKylin Virtualization SIG Maintainer





目录

01

openKylin简介
桌面操作系统开源社区

02

技术创新与开源
Virtio GPU硬件视频加速
框架介绍

03

欢迎加入
openKylin
Virtualization SIG

04

Q&A
互动交流



Part 01

openKYLIN 简介

openKylin 是什么?

openKylin 社区是在开源、自愿、平等和协作的基础上，由基础软硬件企业、非营利性组织、社团组织、高等院校、科研机构和个人开发者共同创立的一个开源社区，致力于通过开源、开放的社区合作，构建桌面操作系统开源社区，推动Linux开源技术及其软硬件生态繁荣发展。

The logo for openKyLin, featuring the word "open" in blue, "Ky" in orange, and "Lin" in blue.

openKylin 的发展



开源



自愿



平等



协作





Part 02

技术创新与开源： Virtio GPU硬件视频加速框架解析

云桌面与GPU虚拟化

对于云桌面应用，我们可以根据图形性能需求将其简单的划分为两大业务场景，一种是以文字编辑为主的普通办公场景，一种是以3D设计、视频编辑等为主的高性能办公场景。目前，针对前者通常采用QXL显卡等，而后者则只能选用vGPU等硬件虚拟化方案，成本大大增加。

为此，openKylin Virtualization SIG选取Virtio GPU作为探索方向，并对其进行了深度优化，大幅提升了3D性能，并增加了硬件视频加速功能，使其成为一种新的、具有更高性价比的云桌面显示方案。

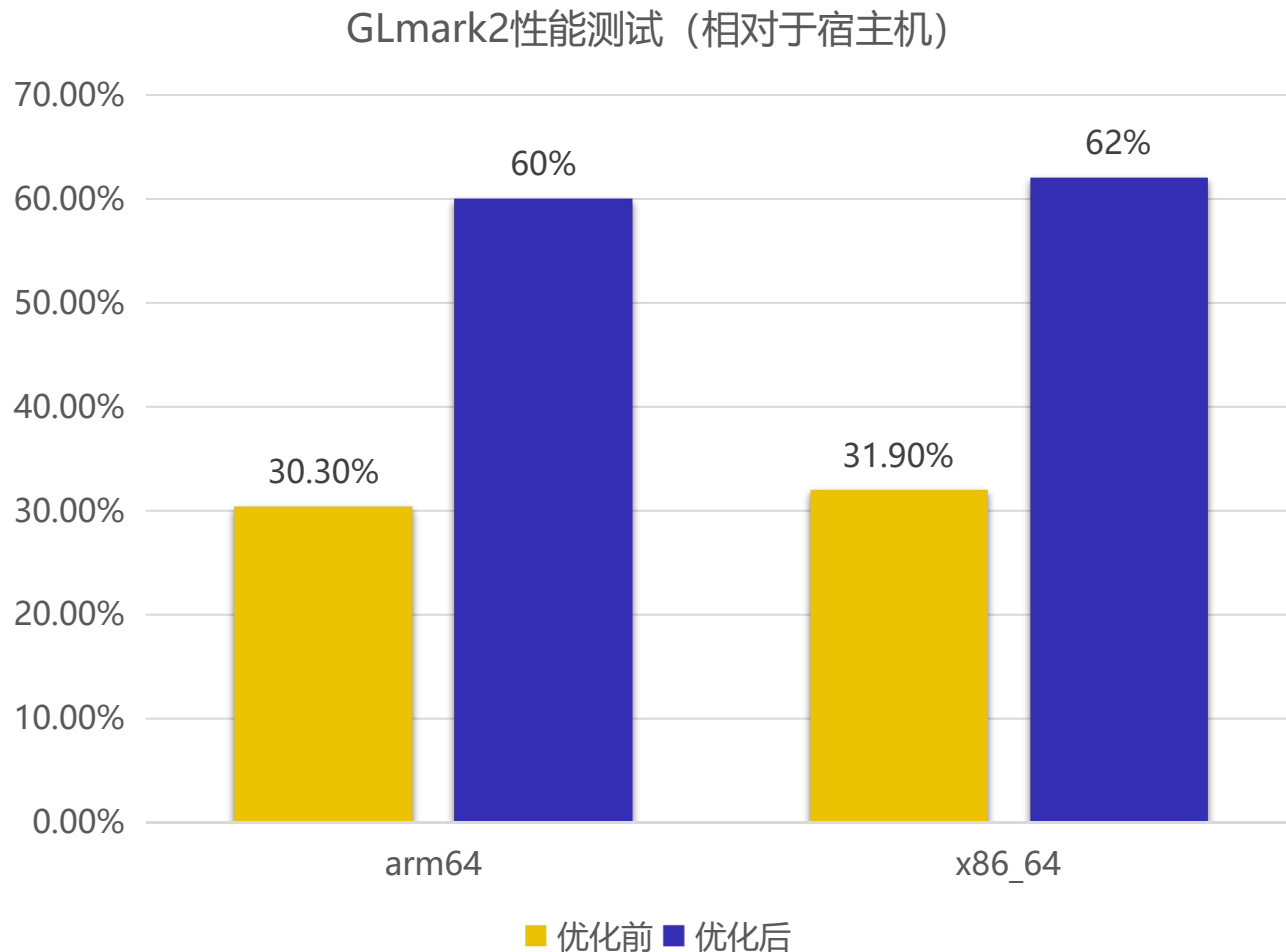
方案	典型代表	优点	缺点	适用场景
软件模拟	QXL	成本低，部署灵活，无需额外硬件	性能低下，无3D加速	普通办公
硬件虚拟化 (透传/分片)	vGPU, SR-IOV	性能非常高，功能完善	成本高、灵活性差，依赖于硬件厂商	AI、高性能计算、GPU服务器、云游戏
API转发	Virtio GPU	部署灵活，性能较强，成本低	软件栈较复杂，性能有一定损耗	高性能办公

Virtio GPU 3D图形性能优化进展

通过使用virglrenderer组件，Virtio GPU可以实现3D加速，完成一些基本的图形处理。但由于其软件栈复杂，性能损耗也较大。

我们尝试对其进行了深度优化，目前其3D性能已有大幅提高。经测试，在arm64和x86_64平台上，虚拟机的3D性能均提高了约一倍。

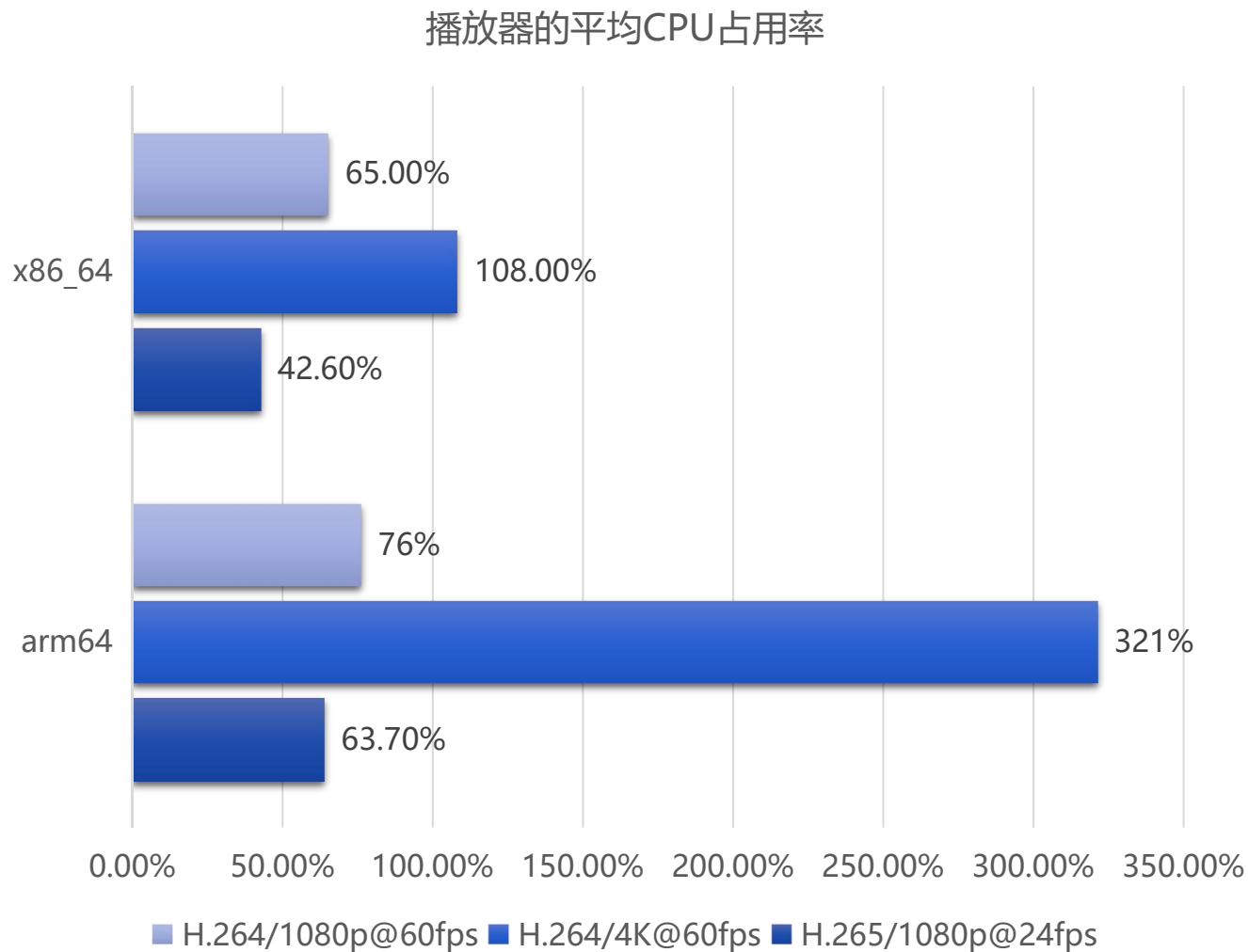
经过上述优化后，Virtio GPU目前已可以应用于一些高性能办公场景，比如简单的3D图形制作、休闲游戏等。



Virtio GPU 视频性能依然不足

虽然3D性能获得了很大提升，但在视频处理方面，Virtio GPU仍存在短板。目前，Virtio GPU由于缺乏硬件视频加速功能，只能使用CPU进行软件编解码。

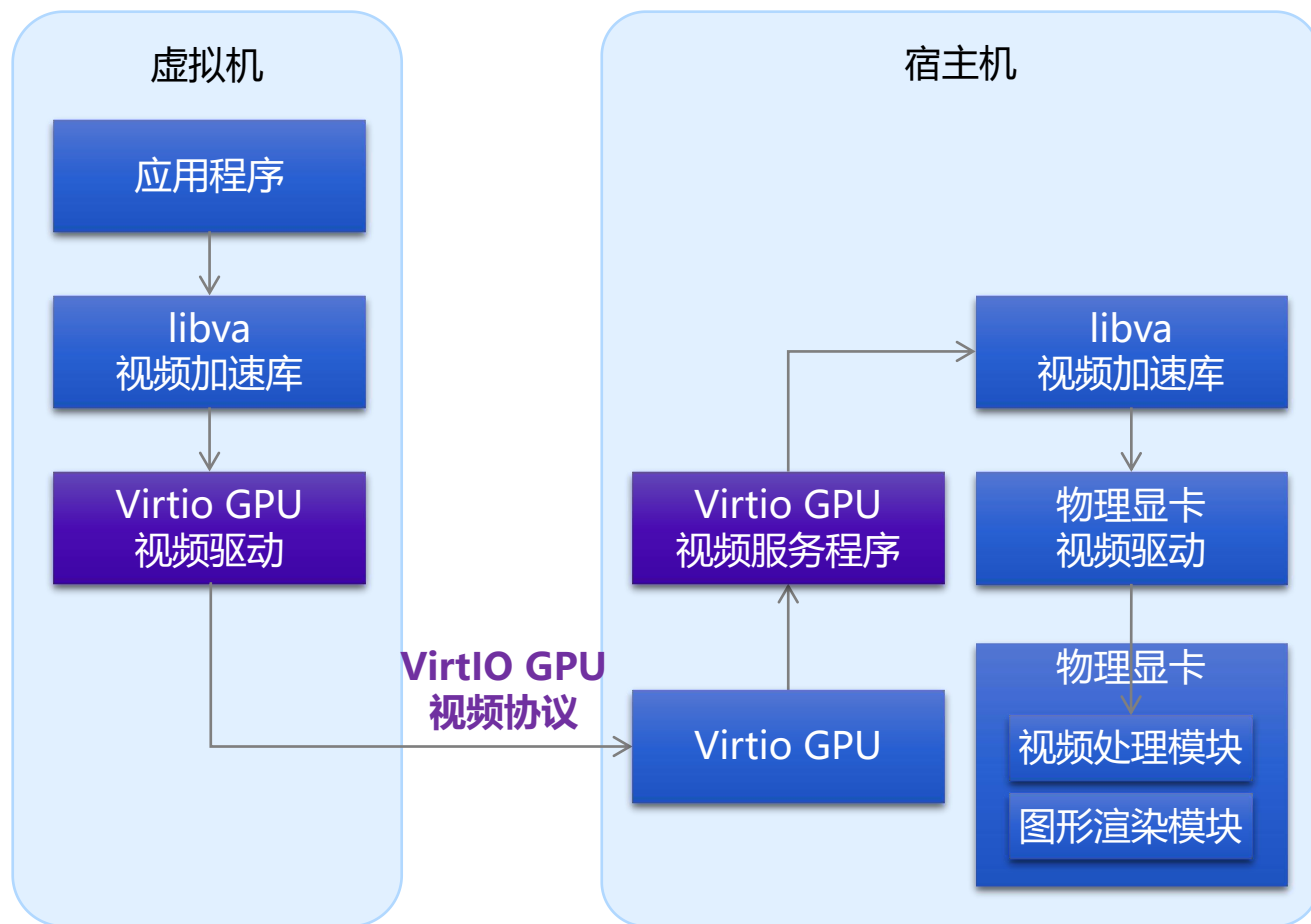
经测试，在arm64平台和AMD Radeon RX550显卡下，在虚拟机内使用MPV播放器播放4K/60fps的H.264视频时，播放器的CPU占用率可达到321%左右，极大的影响了用户的使用体验。



新的Virtio GPU硬件视频加速框架

为了解决Virtio GPU的视频性能问题，我们为其新建了一套硬件视频加速框架。借助该机制，目前已经实现了Virtio GPU下的H.264和H.265的硬件编解码功能。

启用该功能后，由于虚拟机内的播放器在播放视频时，无需再使用CPU进行软件解码，而是借助宿主侧的物理显卡进行硬件解码，因而能极大的降低CPU占用率，提升用户体验。



■ 原有 ■ 新增

硬件视频加速效果

Kylin-Desktop-V10-SP1-x86_64 - QEMU/KVM

文件(F) 虚拟机(M) 查看(V) 发送按键(K)

系统监视器

进程 资源 文件系统 全部进程 搜索

进程名称	磁盘	处理器	进程号	网络	内存	优先级
mpv 媒体播...	0 KB/s	91.3	5080	0 KB/s	350.3 MB	普通

终端

Big Buck Bunny, Sunflower version - mpv

```
[test@Kylin-Desktop-V10-SP1-x86_64 ~]$ mpv Big Buck Bunny, Sunflower version - mpv
[+] Video --vid=1 (*) (h264 1080p60)
[+] Audio --aid=1 (*) (aac 48000Hz stereo)
File tags:
Artist: Blender Foundation
Comment: Creative Commons Attribution-NonCommercial-ShareAlike license
Composer: Sacha Goedege
Genre: Animation
Title: Big Buck Bunny
[mpv] profile : 100 | level : 4
[mpv] fps : 60.000000
A0: [pulse] 48000Hz stereo
V0: [gpu] 3840x2160 yuv4
AV: 00:03:08 / 00:10:34
```

11:13 周二 2022/10/11

无硬件加速

Kylin-Desktop-V10-SP1-x86_64 - QEMU/KVM

文件(F) 虚拟机(M) 查看(V) 发送按键(K)

系统监视器

进程 资源 文件系统 全部进程 搜索

进程名称	磁盘	处理器	进程号	网络	内存	优先级
mpv 媒体播放器	0 KB/s	7.7	18269	0 KB/s	182.4 MB	普通
X	0 KB/s	2.7	979	0 KB/s	44.5 MB	普通

终端

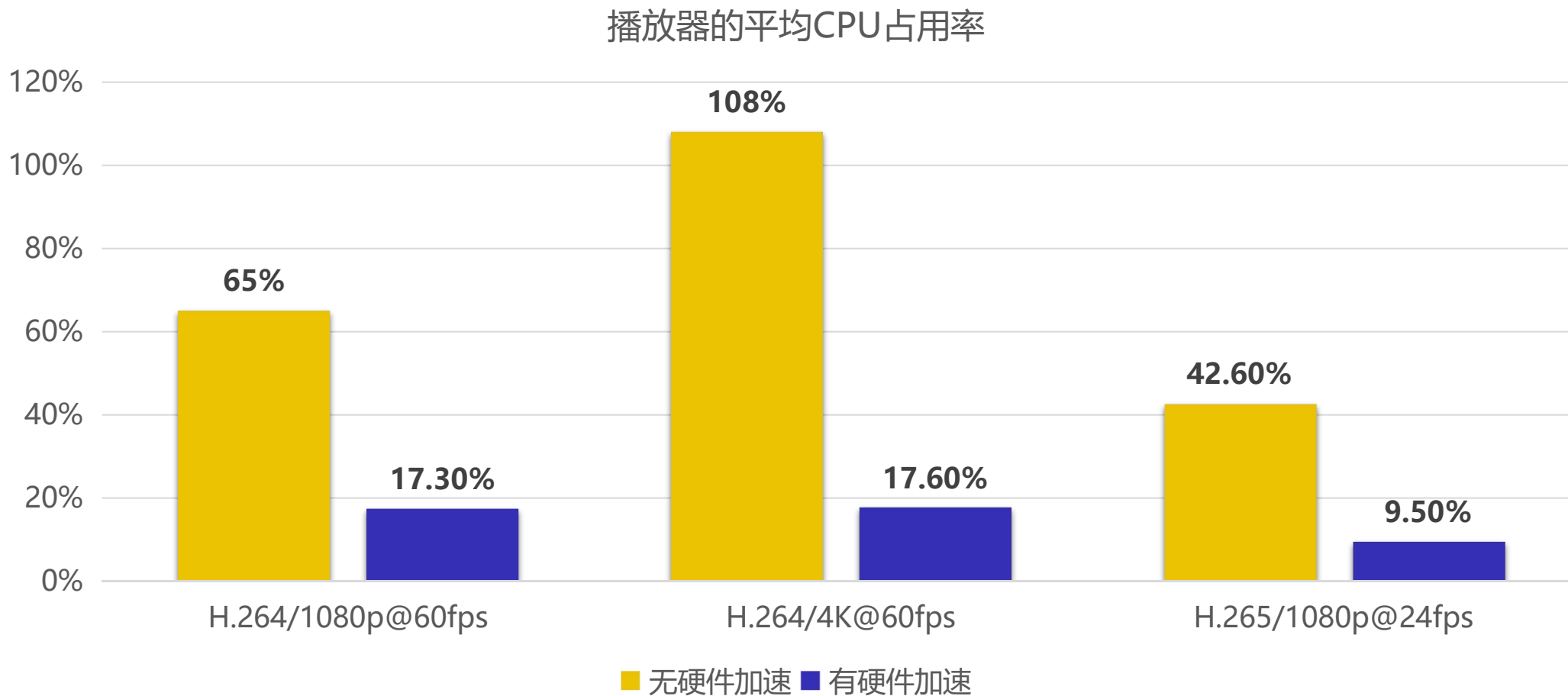
Big Buck Bunny, Sunflower version - mpv

```
[test@Kylin-Desktop-V10-SP1-x86_64 ~]$ mpv Big Buck Bunny, Sunflower version - mpv
[+] Video --vid=1 (*) (h264 1080p60)
[+] Audio --aid=1 (*) (aac 48000Hz stereo)
File tags:
Artist: Blender Foundation
Comment: Creative Commons Attribution-NonCommercial-ShareAlike license
Composer: Sacha Goedege
Genre: Animation
Title: Big Buck Bunny
[mpv] profile : 100 | level : 4
[mpv] fps : 60.000000
A0: [pulse] 48000Hz stereo
V0: [gpu] 3840x2160 yuv4
AV: 00:03:05 / 00:10:34
```

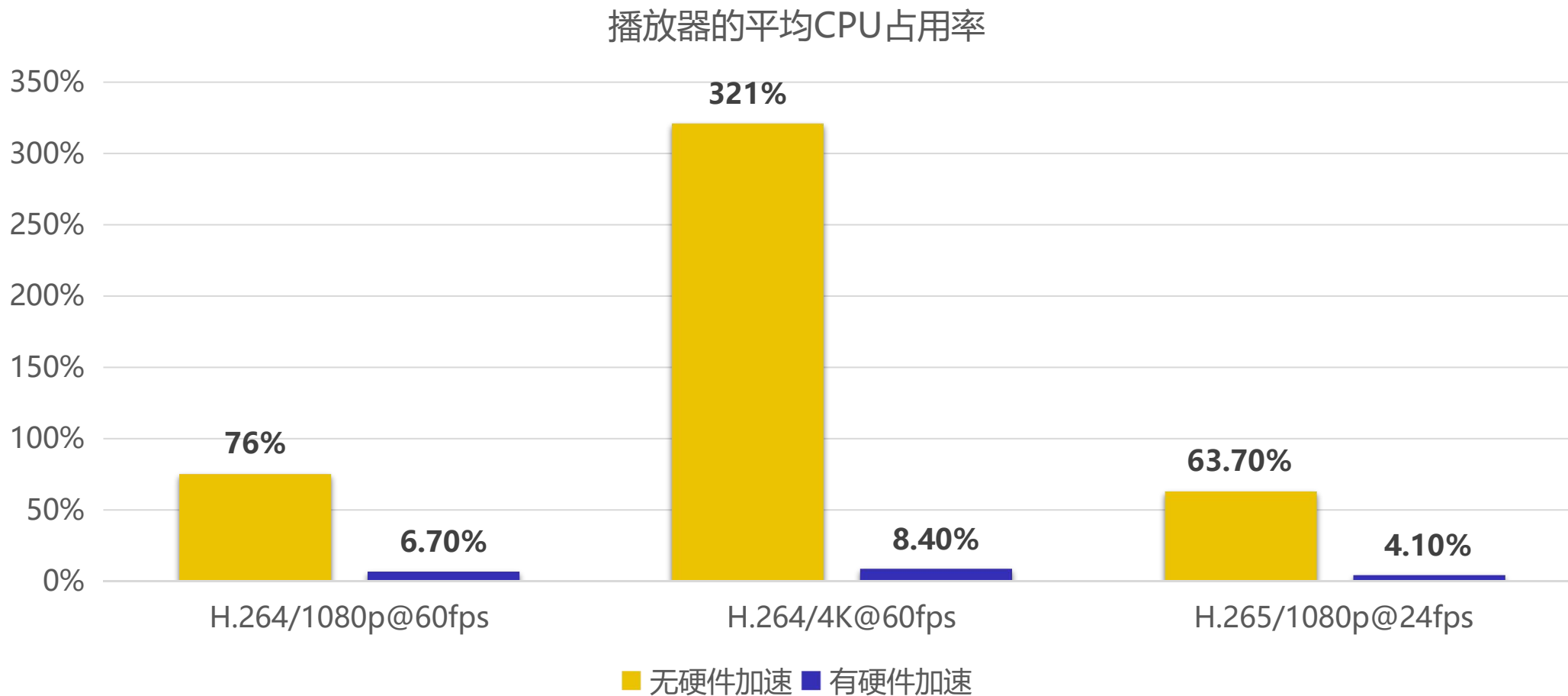
06:11 周一 2022/10/10

有硬件加速

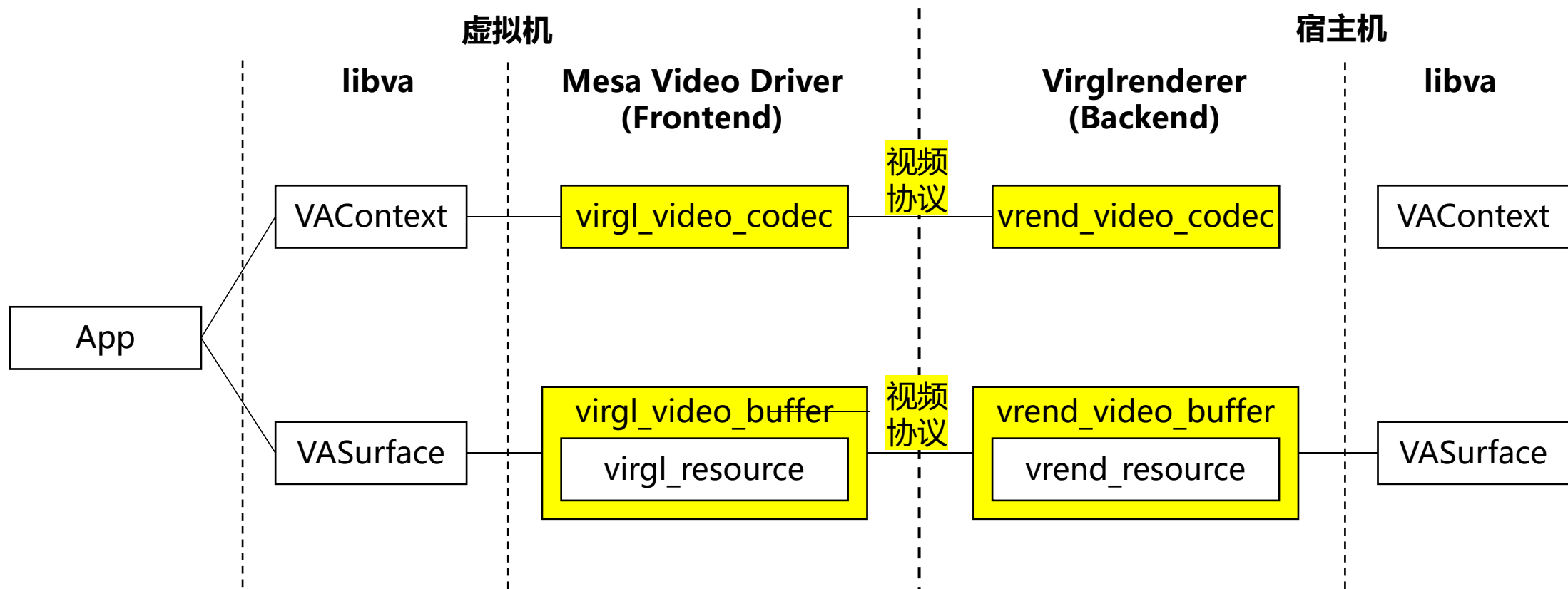
硬件视频加速效果对比 (x86_64)



硬件视频加速效果对比 (arm64)



Virtio GPU硬件视频加速框架



Virtio GPU硬件视频加速框架（续）

Virtio GPU硬件视频加速框架采用前后端设计，总体上分为三部分：

- **Virtio GPU视频前端。** 视频前端作为驱动，负责接收应用程序的视频请求，并将其转化成视频协议，然后发送给后端处理。
- **Virtio GPU视频后端。** 视频后端作为设备，负责接收并解析视频协议，并调用宿主机侧的相关视频加速接口来完成视频请求的处理。
- **Virtio GPU视频协议。** 视频协议是前端和后端之间的通信协议，它定义了创建编解码器、解码比特流等命令及相关数据结构。

在该框架中，定义了两种对象，分别是视频编解码器和视频缓冲区。这两个对象也是前后端结构的，在前端中，它们分别对应着virgl_video_codec和virgl_video_buffer结构，在后端中，它们分别对应着vrend_video_codec和vrend_video_buffer结构。

Virtio GPU视频协议

Virtio GPU视频协议实际上是VIRGL Context Command的扩展，在其基础上新增了创建编解码器、创建视频缓冲区、解码比特流和编码比特流等视频命令。

编解码器是对视频编解码上下文的抽象，在基于VA-API实现的后端中，通常对应着VAContext对象。

视频缓冲区用于存储解码后或编码前的图像数据，在基于VA-API实现的后端中，通常对应着VASurface对象。

解码比特流和编码比特流命令是最核心的编解码命令，绝大部分的编解码操作都是由其完成。

VIRGL Context CMD	说明
VIRGL_CCMD_CREATE_VIDEO_CODEC	创建编解码器
VIRGL_CCMD_DESTROY_VIDEO_CODEC	销毁编解码器
VIRGL_CCMD_CREATE_VIDEO_BUFFER	创建视频缓冲区
VIRGL_CCMD_DESTROY_VIDEO_BUFFER	销毁视频缓冲区
VIRGL_CCMD_BEGIN_FRAME	开始编解码帧
VIRGL_CCMD_DECODE_MACROBLOCK	解码宏块
VIRGL_CCMD_DECODE_BITSTREAM	解码比特流
VIRGL_CCMD_ENCODE_BITSTREAM	编码比特流
VIRGL_CCMD_END_FRAME	结束编解码帧

视频前端的实现

Virtio GPU硬件视频加速框架的前端位于客户机内的mesa中，以用户空间驱动的形式为应用程序提供硬件视频加速接口，如VA-API、VDPAU等。

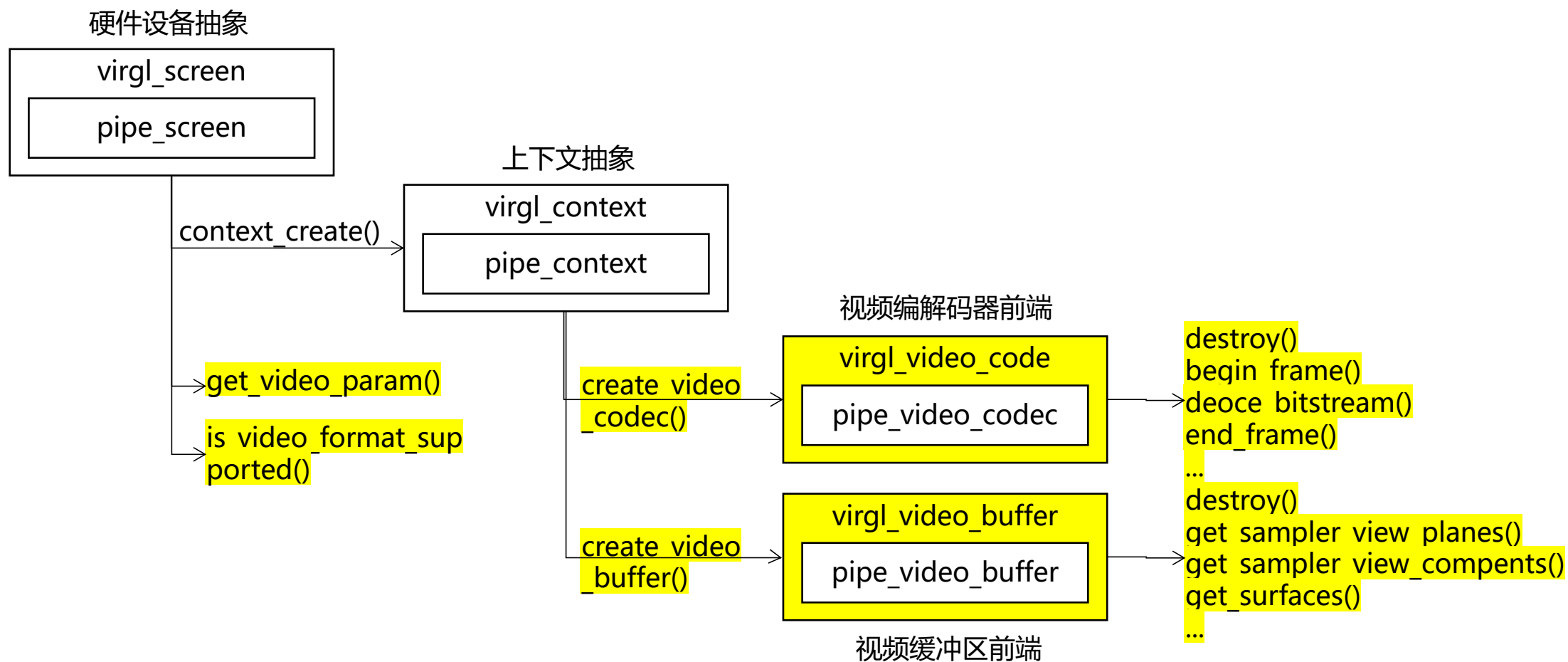
视频前端根据mesa gallium驱动框架，主要实现了以下几个部分：

- virgl_screen对象中的相关方法，主要为视频参数查询等。
- virgl_video_codec对象，表示视频编解码器（或上下文）。
- virgl_video_buffer对象，表示视频缓冲区，用于存储图像数据。

其中，virgl_video_codec继承自pipe_video_codec，并实现了begin_frame, decode_bitstream等方法；virgl_video_buffer继承自pipe_video_buffer，并实现了get_sampler_view_planes等方法。

上述两个对象实例均由virgl_context创建，virgl_context集成自pipe_context，是对Virgl OpenGL等上下文的抽象。

视频前端的实现（续）

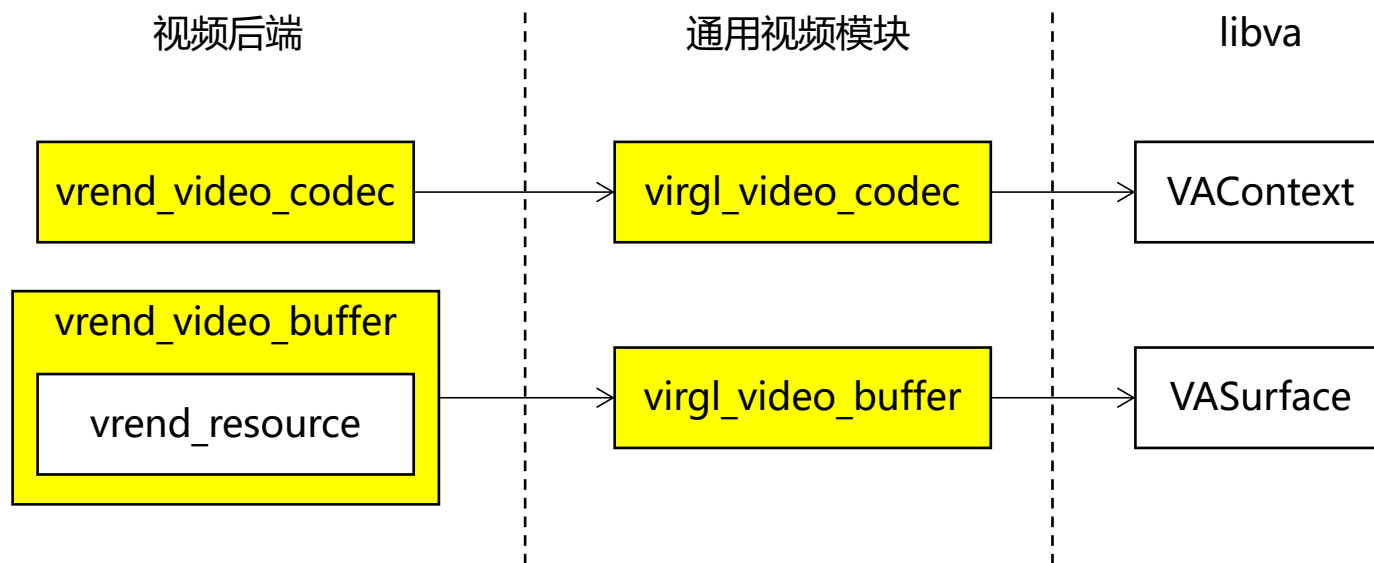


宿主机侧的实现

Virtio GPU硬件视频加速功能的后端位于宿主机侧的virglrenderer组件中，与前端对象相对应的，其定义了两个后端对象，分别为：

- vrend_video_codec：编解码器的后端实现。
- vrend_video_buffer：视频缓冲区的后端实现。

另外，为了支持不同的底层加速接口，采用了通用视频模块来屏蔽接口差异。目前其基于VA-API实现，后续可考虑支持V4L2或NVDEC等私有驱动。



已支持的视频规范

目前，Virtio GPU硬件视频加速框架已支持H.264和H.265视频规范的解码和编码功能（仅支持YUV420图像格式）。

	H.264	H.265	VP8	VP9	AV1	...
编码	√	√	N/A	N/A	N/A	...
解码	√	√	N/A	N/A	N/A	...

开源进展

提交解码补丁

向mesa和virglrenderer
上游提交解码补丁



2022/06

解码补丁被接收

mesa和virglrenderer接
收解码补丁



2022/08

提交编码补丁

向mesa和virglrenderer
上游提交编码补丁



2022/09

编码补丁接收中...

virglrenderer已接收
mesa仍在等待审核...



2022/10

版本要求

对于解码功能，目前virglrenderer已经集成在了0.10.1及以后的版本中，而mesa预计在正式版本22.3.0中集成。

对于编码功能，还在等待上游完全合入，暂时无法直接下载试用。

目前，openKylin社区已经完成里解码和编码功能的合入（yangtze分支），预计后续会集成在openKylin 1.0版本中。

仓库		解码	编码
上游	virglrenderer	≥ 0.10.1	master branch
	mesa	≥ mesa-22.3.0-rc1	N/A
openKylin	virglrenderer	yangtze branch	yangtze branch
	mesa	yangtze branch	yangtze branch

后续工作？

- 支持更多的图像格式，如YUV422，YUV444；
- 支持更多的视频规范，如VP8、VP9、AV1等
- 视频后端增加对NVDEC等私有驱动的支持
- 支持视频的后期处理，如降噪、白平衡等
- ...

Part 03 欢迎加入

openKylin Virtualization SIG

基本信息	https://gitee.com/openkylin/community/tree/master/sig/Virtualization
主要工作	负责openKylin虚拟化相关组件社区技术发展和决策 负责openKylin虚拟化相关软件包的规划、升级和维护 及时响应openKylin虚拟化产品用户反馈和解决虚拟化相关问题
主要项目	kernel, qemu, libvirt, edk2, virglrenderer, spice, mesa, ...
邮件列表	virtualization@lists.openkylin.top



Part 04 Q & A



openKylin
微信公众号



openKylin
微信小程序



openKylin
微博



openKylin
bilibili

感谢观看



云原生社区公众号



云原生社区长沙站