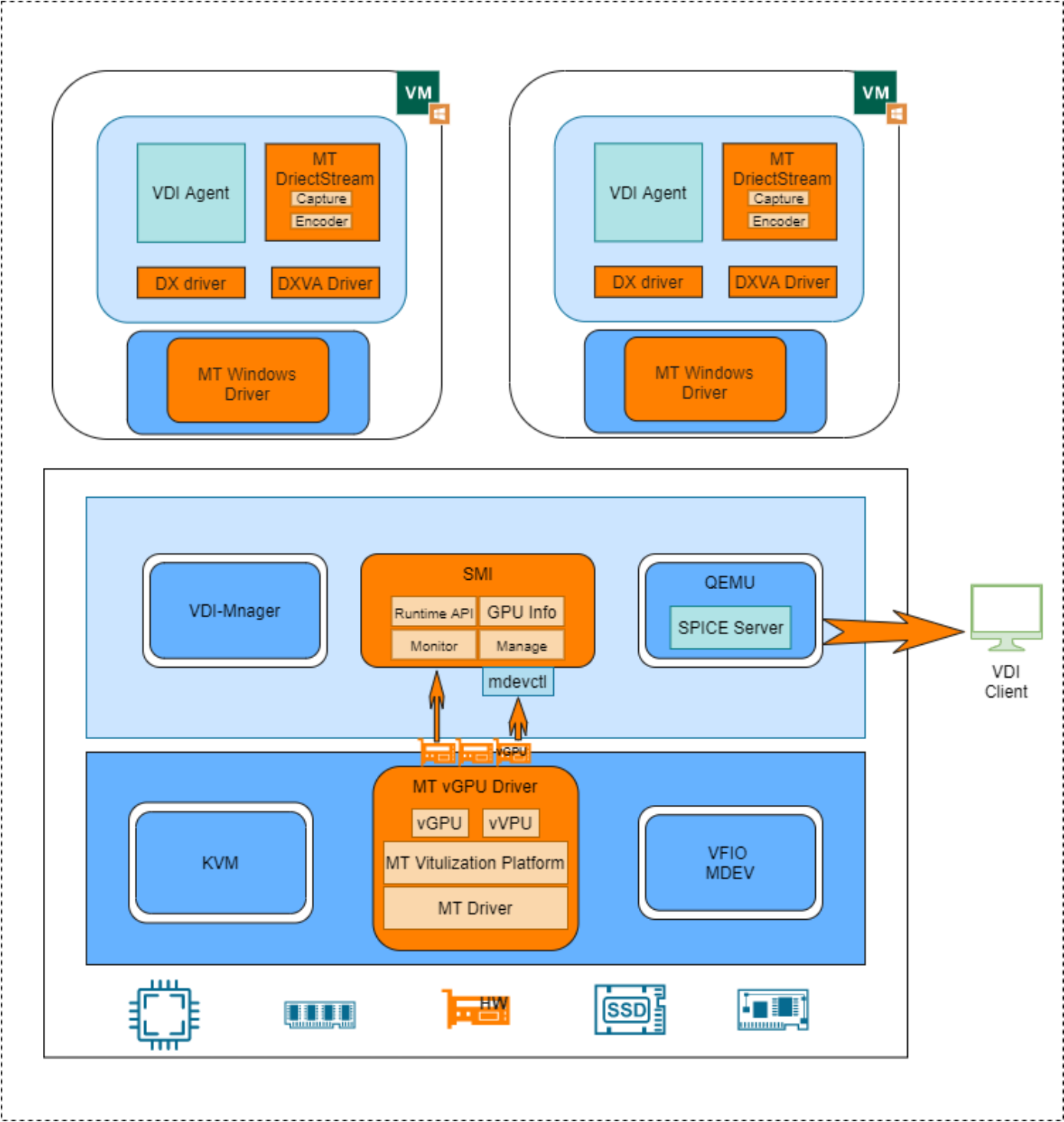


S2000 MT vGPU 2.3.0驱动发布手册

MT vGPU简介

MT vGPU是摩尔线程GPU虚拟化的整体产品名称。摩尔线程GPU作为重要的算力，软件开发并提供了GPU虚拟化技术，满足数据中心中以GPU直通、vGPU的方式使用GPU的需求。摩尔线程通过PCIe Passthrough技术支持GPU直通，通过软件截获直通的方式开发了vGPU驱动，并在后续产品中会支持基于SR-IOV方案。整体的软件产品通过MT vGPU的软件产品形态提供。虚拟化出来的vGPU的功能与物理GPU设备形态上完全一致，向上可以支持虚拟机中的业务运行。虚拟化系统的架构图如下图所示：



名词解释

- Host：服务器主机
- Guest：虚拟机
- vGPU：虚拟GPU

MT vGPU驱动安装及使用方法

MT vGPU驱动软件包

软件包名称	软件包描述
MT_vGPU_HOST_v2.x.x_For_	GPU虚拟化驱动，Host端使用，驱动GPU，可将物理GPU切分成多个虚拟GPU（vGPU），并发支持多个虚拟机使用
MT_vGPU_DKMS_v2.x.x	适用DKMS的deb包和rpm包，Host端使用
MT_vGPU_GUEST_v2.x.x	GPU Guest驱动程序，Guest端使用，驱动vGPU，使能GPU的渲染、编解码功能
MT_vGPU_PES_v2.x.x	GPU Guest驱动安装包，Guest端使用
MT_vGPU_MTML_v1.x.x	GPU管理函数库，Host端使用
MT_vGPU_GMI_v1.x.x	GPU监控、管理工具，Host端使用
MT_vGPU_DirectStream_v2.x.x	屏幕抓取及GPU硬件编码SDK，在Windows Guest中工作，调用GPU的硬件能力对数据进行编码
MT_vGPU_MTAPI_v2.x.x	virtual display 自定义分辨率设置工具，Guest端使用

准备工作

服务器BIOS设置

- 超4G地址空间译码：开启；详细设置方法参考对应服务器操作手册
- PCIe版本：PCIe Gen3/Gen4；详细设置方法参考对应服务器操作手册
- VMX（Intel）/SVM（AMD）：开启；详细设置方法参考对应服务器操作手册

PCIe设备基本信息查询

- 为确保服务器和操作系统已经识别 MTT S系列GPU，在 Host 操作系统中输入 lspci 命令查询设备 ID
MTT S2000
\$ lspci -d 1ed5:
03:00.0 3D controller: Device 1ed5:0123
04:00.0 3D controller: Device 1ed5:0123

正常情况下，对应显卡的显示内容应该如上所示

驱动安装

拷贝依赖包和firmware

操作：将 MT_vGPU_HOST_v2.x.x_For_ 驱动包拷贝至 Host 服务器，解压后，将 firmware 中固件文件拷贝至 /lib/firmware/mthreads/

```
$ mkdir -p /lib/firmware/mthreads/  
$ cd MT_vGPU_HOST_v2.0.0_For_xxx  
$ cp ./firmware/* /lib/firmware/mthreads/
```

检验：通过命令行查看并确认文件存在

```
$ ls -l /usr/lib/firmware/mthreads/  
total 3408  
-rw-r--r--. 1 root root 133120 Nov 6 15:59 mtvpu-00-1.0.bin  
-rw-r--r--. 1 root root 1286976 Nov 6 15:59 mtvpu-01-1.0.bin  
-rw-r--r--. 1 root root 1292864 Nov 6 15:59 mtvpu-01-2.0.bin  
-rw-r--r--. 1 root root 189440 Nov 6 15:59 mtvpu-02-1.0.bin  
-rw-r--r--. 1 root root 433408 Nov 6 15:59 mtvpu-02-2.0.bin  
-rw-r--r--. 1 root root 41632 Nov 6 15:59 musa.fw.1.0.0.0.vz.win  
-rw-r--r--. 1 root root 46272 Nov 6 15:59 musa.fw.1.1.0.0.vz.win  
-rw-r--r--. 1 root root 46304 Nov 6 15:59 musa.fw.1.2.0.0.vz.win
```

安装Host端显卡驱动

操作 1. 拷贝文件至内核目录

```
$ cp mtgpu.ko /lib/modules/$(uname -r)/
```

2. 导出依赖列表

```
$ depmod -a
```

3. 安装 mtgpu

```
$ modprobe mtgpu
```

验证 1. 列出所有当前加载的包含内核模块，过滤输出，只显示包含 mtgpu 字符串的行

```
$ lsmod | grep "mtgpu"  
mtgpu 1634304 0  
mdev 24576 2 mtgpu,vfio_mdev  
drm_kms_helper 184320 3 mtgpu  
kvm 663552 2 mtgpu,kvm_intel  
drm 491520 3 mtgpu,drm_kms_helper
```

看到上述内容表示驱动已经正常安装。2. 显示 Vendor ID 为 1ed5 的设备信息

```
# MTT S2000
$ lspci -vd 1ed5:
03:00.0 3D controller: Device 1ed5:0123
...
Kernel driver in use: mtgpu
Kernel modules: mtgpu
04:00.0 3D controller: Device 1ed5:0123
...
Kernel driver in use: mtgpu
Kernel modules: mtgpu
```

如果 lsmod 命令的输出包含 mtgpu 内核模块，且 lspci 命令查询结果显示 MT GPU 设备正在使用 mtgpu 模块 作为驱动，则表示安装成功。

vGPU切分

vGPU的配置推荐支持三种方式

- mdevctl工具
- 直接操作设备文件
- mthreads-gmi(摩尔线程GPU管理工具)

三种方式的本质都是对设备文件进行操作以达到对vGPU 设备管理的目的。MT vGPU支持的显卡切分类型请见附录 4.1 MTT S系列显卡支持的vGPU切分类型

切分方式一：推荐使用mdevctl工具对显卡进行切分操作，使用 mdevctl 工具切分显卡的流程如下：

1. 查看GPU地址

```
$ lspci -d 1ed5:
03:00.0 3D controller: Device 1ed5:0123 # 03:00.0 为第一颗GPU的地址
04:00.0 3D controller: Device 1ed5:0123 # 04:00.0 为第二颗GPU的地址
```

2. vGPU启用

```
$ mdevctl start -u c4f702ca-c69d-4d7d-a526-5fdcf78d3428 -p 0000:03:00.0 --type mtgpu-1101
```

- c4f702ca-c69d-4d7d-a526-5fdcf78d3428 是UUID，是vGPU设备的唯一标识；此ID可以是固定的ID，也可以通过uuidgen工具生成： uuidgen => xxxxxxxx
- -p 之后跟进的是使用的GPU芯片地址
- -type 描述的是生成vGPU所使用的vGPU类型

1. 查看创建的vGPU设备

```
$ mdevctl list
c4f702ca-c69d-4d7d-a526-5fdcf78d3428 0000:03:00.0 mtgpu-1101
```

切分方式二：操作设备文件

1. 查看并操作文件设备

```
$ find /sys/ -name mdev_supported_types
/sys/devices/pci0000:00/0000:00:01.0/0000:01:00.0/0000:02:00.0/0000:03:00.0
/sys/devices/pci0000:00/0000:00:01.0/0000:01:00.0/0000:02:08.0/0000:04:00.0

$ echo c4f702ca-c69d-4d7d-a526-5fdcf78d3428
/sys/devices/pci0000:00/0000:00:01.0/0000:01:00.0/0000:02:00.0/0000:03:00.0/
mdev_supported_types/mtgpu-1101/create
```

2. 查看切分出的vGPU

```
$ ls /sys/bus/mdev/devices/
c4f702ca-c69d-4d7d-a526-5fdcf78d3428
```

安装Guest端显卡驱动

创建带有MT vGPU的QEMU虚拟机

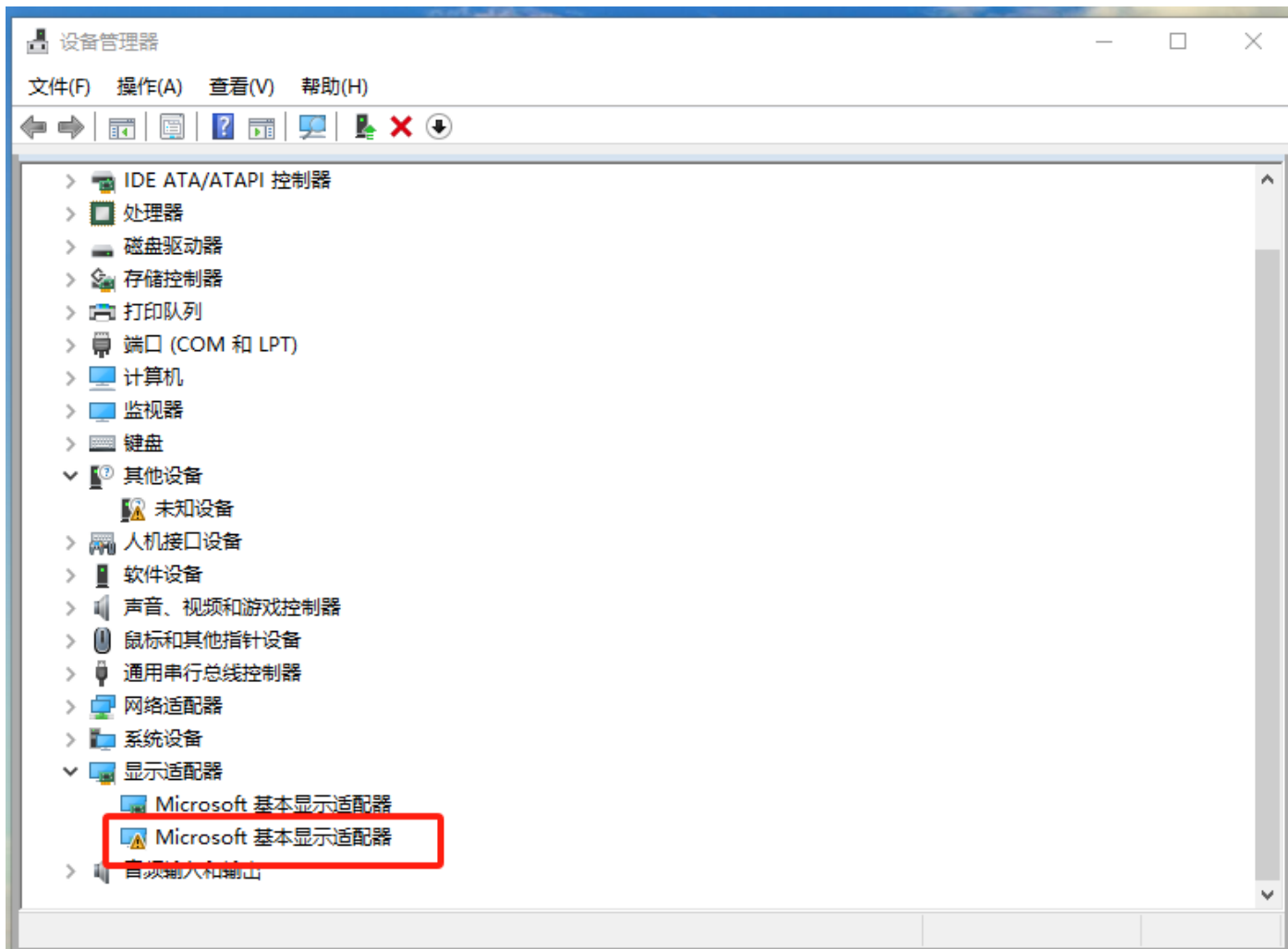
使用QEMU命令行的参考命令如下：

```
sudo qemu-system-x86_64 -name 'Moore Threads GPU' \
-m machine q35,accel=kvm,usb=on \
-cpu host -smp 2,sockets=1 -m 4G \
-hda win10-vdi-snapshot.qcow2 \
-usb -device usb-tablet \
-device vfio-pci,sysfsdev=/sys/bus/mdev/devices/c4f702ca-c69d-4d7d-a526-5fdcf78d3428 \
-vnc :01 \
-net user -net nic \
-device intel-hda \
-device hda-duplex
```

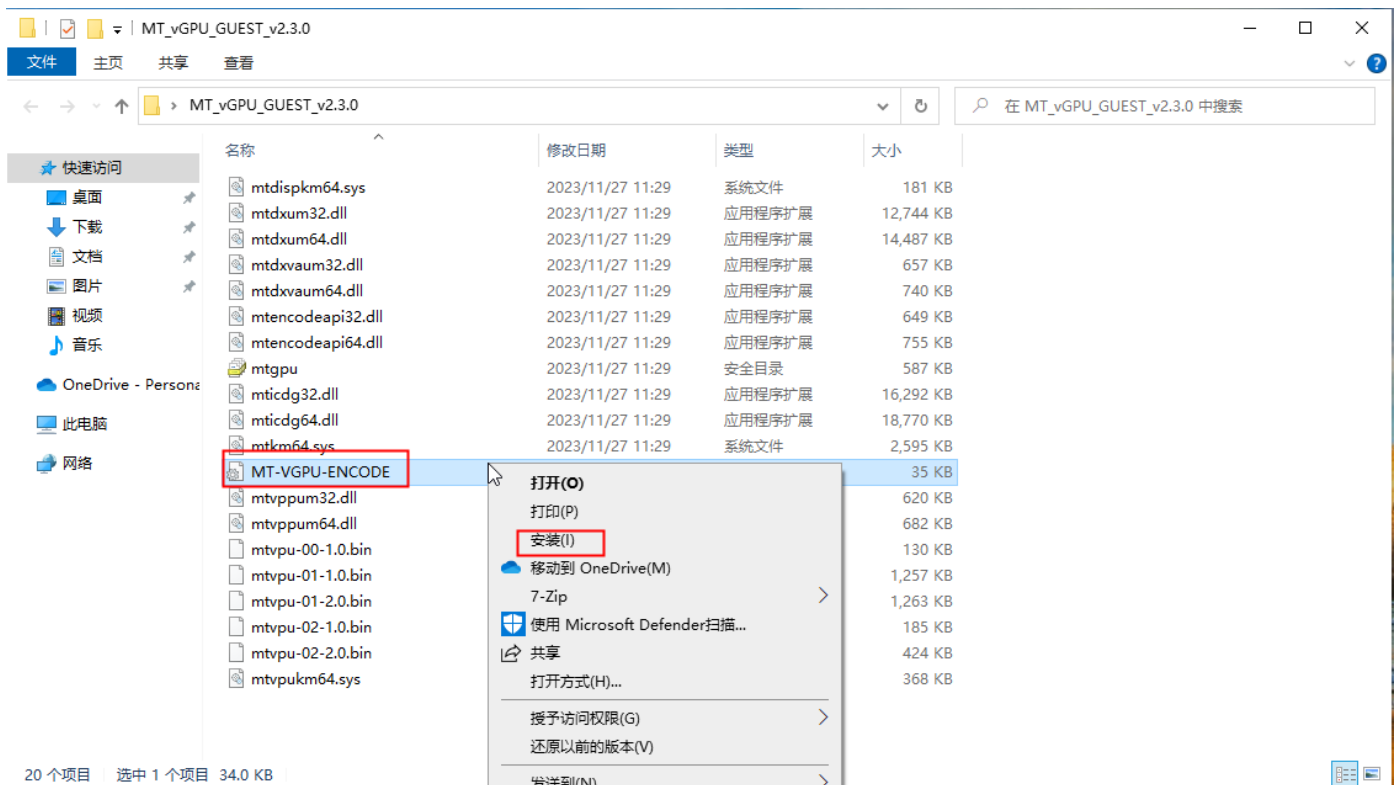
其中，-device vfio-pci,sysfsdev=/sys/bus/mdev/devices/c4f702ca-c69d-4d7d-a526-5fdcf78d3428 为增加的vGPU的参数

在虚拟机中安装Windows 10 vGPU驱动

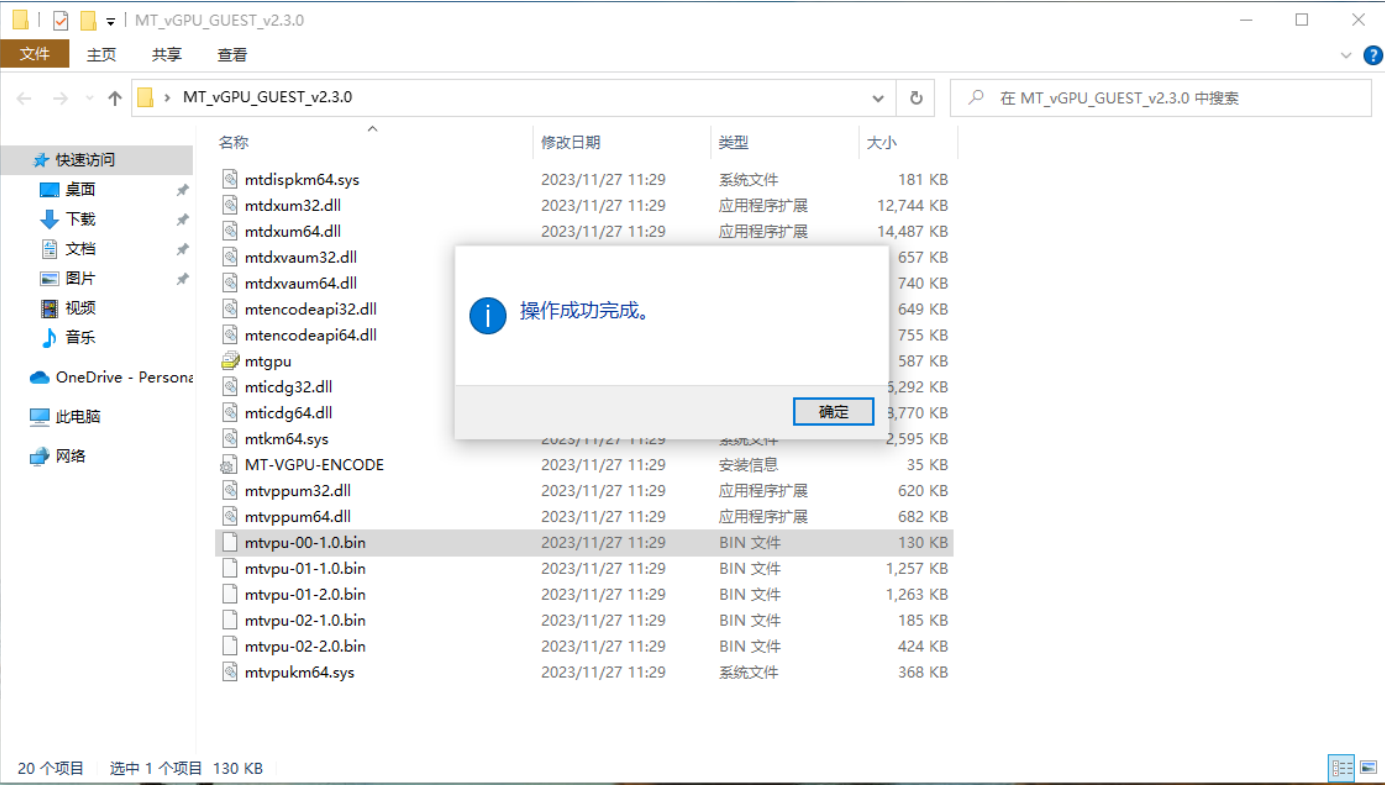
vGPU 需要安装 Windows 驱动才能作为 3D 加速设备进行工作。未安装摩尔线程 Windows 驱动的 vGPU 设备如下图所示：



在虚拟机中，安装Windows10 vGPU驱动，右键单击INF文件，从菜单中选择安装：



安装成功，如下图所示：



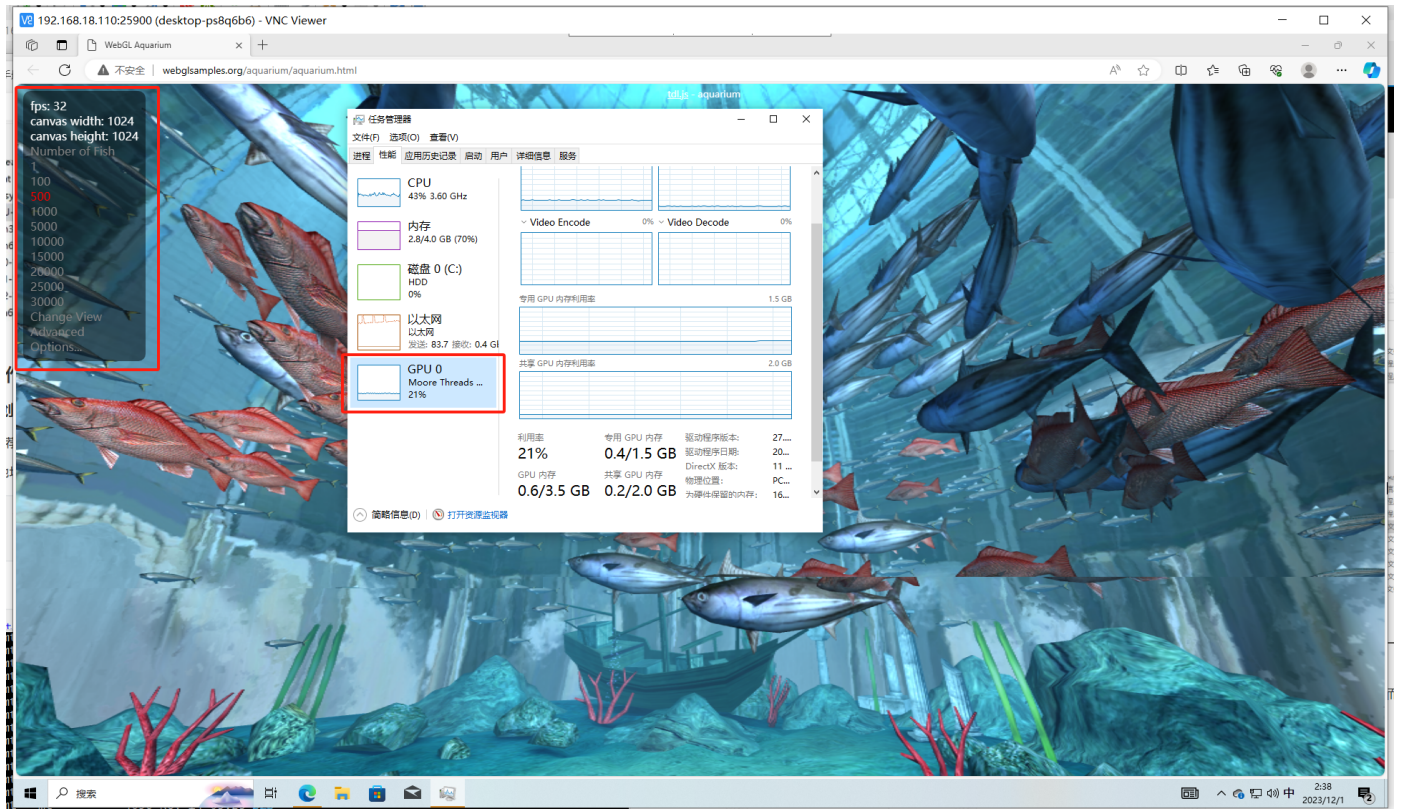
使用MT Virtual Display

屏幕设置成仅在2上显示



使用vGPU 运行webgl

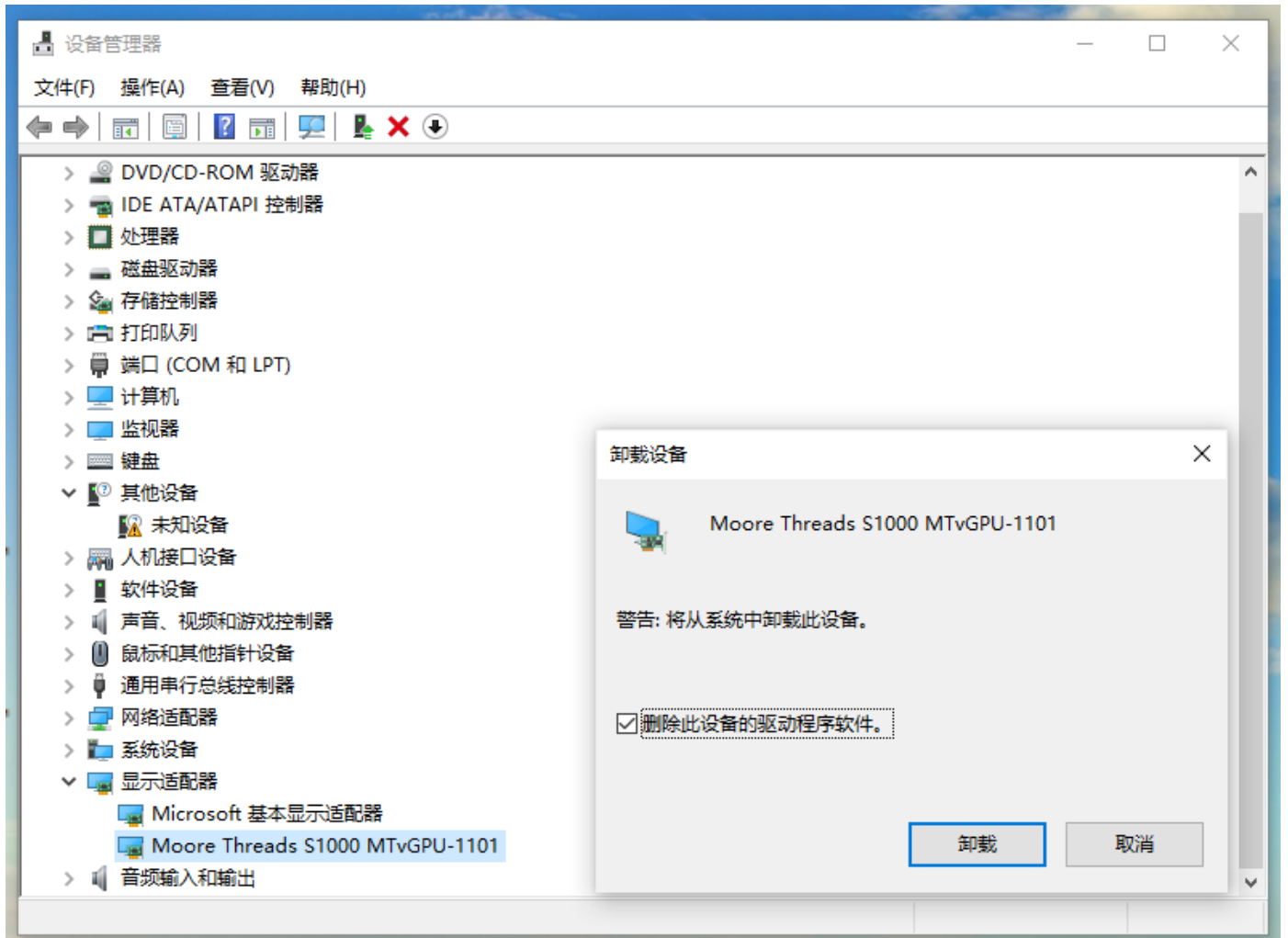
使用MTT S2000 加速webgl渲染，可以看到CPU利用率有下降而且webgl 渲染帧率也能稳定在30fps



驱动卸载

Guest驱动卸载

在Guest设备管理器中选择卸载设备，并勾选"删除此设备的驱动程序软件" 即可。



Host驱动卸载

先通过 `lspci -d 1ed5:` 命令获取到所有moore threads GPU设备的<b.d.f>信息，再逐个使用以下命令将其解 绑：

```
$ echo "0000:<b.d.f>" > /sys/bus/pci/devices/0000:<b.d.f>/driver/unbind
```

最后使用下面命令删除mtgpu驱动。

```
$ rmmod mtgpu
$ rm /usr/lib/modules/$(uname -r)/mtgpu.ko
```

说明：卸载驱动前请确保所有 VM 处于关机状态。

驱动升级

摩尔线程 vGPU HOST和GUEST驱动互相匹配，需要同步升级：

1. 首先参照 "2.4.1 Guest驱动卸载" 卸载旧的GUEST驱动；
2. 参照 "2.4.2 Host驱动卸载" 卸载旧的HOST驱动；
3. 重启 HOST设备；
4. 参照 "2.3 驱动安装" 重新安装新驱动。

MT GPU管理工具使用

通过摩尔线程 GPU 管理接口工具 mthreads-gmi 可以实现 GPU 和 vGPU 管理和监控。显示帮助信息

```
$ mthreads-gmi --help
```

监控 GPU 详细信息

```
$ mthreads-gmi -q -d MEMORY,UTILIZATION,TEMPERATURE,CLOCK -l 10
```

更多命令，参见 mthreads-gmi 命令参考。