

GPU 虚拟化技术及应用研究

全伯兵¹, 杨昕吉², 谢振平¹, 王士同¹, 梅向东²

(1. 江南大学 数字媒体学院, 江苏 无锡 214122; 2. 赞奇科技发展有限公司, 江苏 常州 213000)

摘要: GPU 即图形处理器, 在构建高性能计算系统中占有重要地位。为更好地利用 GPU 的计算优势, GPU 虚拟化技术受到关注。系统介绍了 GPU 虚拟化技术, 并对独占使用、设备模拟和应用层接口等 3 种虚拟化技术进行了研究。从虚拟化桌面应用、GPU 通用计算虚拟化和 GPU 虚拟化的安全问题等方面探讨了 GPU 虚拟化研究的需求及目标, 给出了 GPU 虚拟化市场应用的可行方案。

关键词: GPU 虚拟化; 虚拟化桌面; 通用计算虚拟化

DOI: 10.11907/rjdk.151107

中图分类号: TP393

文献标识码: A

文章编号: 1672-7800(2015)006-0153-04

1 GPU 虚拟化研究意义

GPU 主要实现计算机的图形呈现, 具有高性能的多处理器阵列, 随着 GPU 技术的发展, 其在 3D 渲染和大规模并行计算中起着越来越重要的作用^[1]。

虚拟化技术^[2]是对传统计算资源使用方式的一次新突破, 虚拟计算技术的引入打破了真实计算中软件与硬件之间的紧耦合关系, 为不同需求的计算系统架构带来了活力支撑, 图 1 给出了计算系统虚拟化的常见分类。目前市面上的 I/O 设备有成千上万种, I/O 设备的复杂性、多样性以及封闭性等特点使其成为虚拟化技术发展的瓶颈, 导致硬件资源不能得到充分利用, 使虚拟机环境性能下降, 甚至在某些情况下发生功能缺失现象。GPU(图形处理器)是显卡的主要组成部分, 功能包括图形(2D 和 3D)计算与通用计算两部分。目前除去少数 2D 显示标准外, 国际上对显卡的硬件电气接口并没有统一的标准和规范, 各主要硬件生产厂家之间设备接口互不兼容。由于技术革新换代等原因, 同一厂家的不同产品之间兼容也不好。另外, 出于商业利益, 这些设计标准都是厂家的核心机密, 从不对外公布设计细节和源码, 使得在操作系统层面适应不同硬件的驱动协议栈开发目标难以实现。目前商业级的 VMware^[3] 以及开源的 Xen^[4] 等主流虚拟机管理器(VMM)都只在小范围内解决了 2D 图形显示问题, 很少涉及通用计算和 3D 加速这两项 GPU 主要功能。在 I/O 虚拟化研究中, 当设备虚拟化不能在硬件电气层实现时,

通常采用在逻辑层对其进行模拟的方式, 以网卡虚拟化为例, 就是在传输层或网络层完成的虚拟化。然而, 在大规模数值计算时, GPU 的计算效率是 CPU 的成百上千倍, 这是 CPU 无法企及的。基于上述种种原因, 迄今为止 GPU 还难以真正虚拟化。

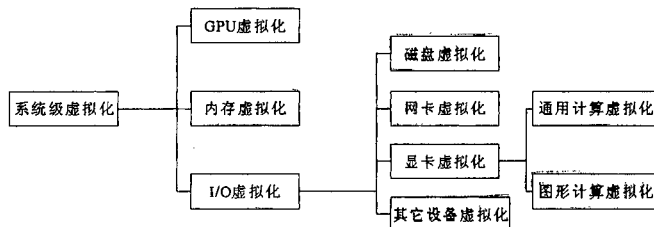


图 1 计算系统虚拟化分类

2 GPU 虚拟化方法

目前 GPU 虚拟化技术有 3 类: ①独占使用; ②设备模拟; ③应用层接口虚拟化(API 重定向法), 下面分别展开阐述。

2.1 独占使用

独占使用(VMM pass-through)是指绕过虚拟机的管理系统, 将 GPU 使用权直接分配给某一虚拟机, 任何时刻都只有一台虚拟机拥有 GPU 的使用权限。这种虚拟化方式使 GPU 的独立性与完整性得以保存, 而且效果与非虚拟条件下的结果差别不大, 同时, 能够进行通用计算。然而, 这种方式与虚拟机动态分配、资源共享的理念不符。

作者简介: 全伯兵(1989—), 男, 山东菏泽人, 江南大学数字媒体学院硕士研究生, 研究方向为机器学习、数据挖掘、数字图像处理; 谢振平(1979—), 男, 江苏常州人, 江南大学数字媒体学院副教授, 研究方向为数据挖掘、数字图像处理; 梅向东(1966—), 男, 湖北武汉人, 赞奇科技发展有限公司高级工程师, 研究方向为多媒体及计算机应用。

另外,独占使用方案兼容性不理想。以 Xen VGA Passthrough^[5]为例,它运用了英特尔设备虚拟^[6](Intel VT-d)技术,在运行时将主显设备直接分配给某一客户虚拟机,使宿主虚拟机和其它客户虚拟机都没有访问该 GPU 的权限。同时,VESA 模式、VGABIOS、IO 端口、文本模式、内存映射这些显卡的特殊细节在客户虚拟机中实现并且能够直接访问。使用 Xen VGA Passthrough 技术的 GPU 执行效率高、功能全,但它不具有设备复用功能,只能被单一系统独占式使用,因而在实际中较少应用。

独占设备法的设计思路是使原生驱动和硬件能够被客户操作系统直接使用,缺少必要的中间层维护和 GPU 状态跟踪,对实时迁移等高级特性支持不足。比如 VMware 在用户文档中明确指出:开启 VMDirectPath I/O 功能时,其对应的虚拟机将会失去执行实时迁移、挂起/恢复的能力,Xen Passthrough 则禁止执行 restore/ save/migration 等操作。

2.2 设备模拟

该技术是使用 CPU 来模拟 GPU 的电气化接口,从而实现 GPU 的功能。基于设备模拟的 GPU 虚拟化在没有物理硬件的支持下,通过纯软件的方式实现 GPU 虚拟化。由于 GPU 电气单元数量庞大,出于商业利益考虑,各大厂家的 GPU 电气接口细节对外是保密的,并且不同的生产厂家甚至同一厂家不同型号的 GPU 结构差异也很大,这在很大程度上增加了设备模拟的难度。这种模拟器只是在软件调试和硬件设计时得到应用,并不能作为 VMM 中的虚拟设备使用。QEMU^[7]目前具备基本 2D 功能的显卡(Cirrus CLGD 5446 PCI VGA card),已经在虚拟化系统广泛应用,但也只仿真了 GPU 的部分功能,如在 KVM 和 Xen 中负责虚拟化显示设备。VMware SVGAII^[8]是 VMware 中独立实现的仿真设备,能支持 2D 显示和 Direct3D 接口,但不支持 OpenGL,且对 Direct3D 的支持仍然是使用 API 重定向技术。

由于在设备模拟中,设备的状态位于内存、CPU 等 VMM 管理部分,其对虚拟机的某些高级特性如实时迁移等适应性较好,虚拟机的实时迁移、系统快照和多路复用等功能均能拓展到 GPU 相关领域。然而使用 CPU 模拟方式并不能高效完成 GPU 的并行计算,在执行时间上 CPU 是 GPU 的成百上千倍,导致实用价值较差。

2.3 API 重定向技术

应用接口虚拟化(API remoting, API 重定向技术),是指对 GPU 相关的应用程序编程接口在应用层中进行拦截,然后使用重定向(使用 GPU)或模拟(不使用 GPU)方式实现相应功能,将完成的结果返回给对应的应用程序。目前,OpenGL 等传统图形 API 中已经广泛使用了 API 重定向虚拟化技术。

Chromium^[9]提出了一种将特定的模块插入到 OpenGL 指令流中的机制,改变了 OpenGL 的执行流程,从而实现了大规模分布式图形呈现与渲染。Blink^[10]项目是采用拦截 OpenGL 在多个客户操作系统之间的复

用 3D 显示设备。VMware 的 GPU 虚拟化方案则更加复杂,它结合 API 重定向技术与设备仿真两种方案特点,采用前端—后端的运行模式。前端由虚拟 GPU 与伪显卡驱动构成,位于客户操作系统中;后端则由原生显卡驱动和特殊线程 MKS(Mouse—Keyboard—Screen)构成,位于宿主操作系统中。当 GPU 的相关指令传递到 MKS 时,MKS 会按照指令类型驱动不同的功能单元,如 2D 图形指令对应 X11 或 GDI 端口,3D 图形指令对应 Direct3D 或 OpenGL 端口。前后端之间的通信采用 FIFO 管道与 DMA 两种传输模式,VMware 通过设置共享内存区域进行虚拟显存,优化了客户操作系统与宿主操作系统的数据传输性能,减少了数据复制操作的次数。

API 重定向技术方案处理参数在 API 层进行,能够管理虚拟机内部状态。VMware、Blink 等方案对虚拟机的高级特性有不同程度的支持,如 VMware 实现了针对 2D、3D 应用的快照、复用、冻结、迁移等功能,完全支持虚拟机的各种高级特性,VMGL^[11]实现了 3D 应用在虚拟机中的冻结/恢复与多路利用功能。

通过分析,可将上述 3 种 GPU 虚拟化方法汇总为表 1。API 重定向研究在通用计算与图形计算方面成果显著,但性能问题依然是不足之处。在将本地执行代码重定向到远程虚拟机执行时会涉及大量数据的传输和复制,这是各种虚拟方案难以避免的问题,而采用域间通信技术是解决此问题的一种有效方法。

表 1 GPU 虚拟化现状

虚拟化方法	图形计算虚拟化	通用计算虚拟化	对虚拟及高级特性支持
独占设备	Xen、VGA Pass-through、Parallels Desktop、VMDirect-Path I/O	Xen、VGA Pass-through、Parallels Desktop、VMDirect-Path I/O	不支持
设备模拟	QEMU、VirtualGPU	Barra(模拟器)、GPG-PU—Sim(模拟器)	仅支持图形计算的高级特性
API 重定向	VMGL、BLINK、Parallels Desktop	VCUDA、GVIM、gVirtus	支持图形和通用计算的高级特性

3 GPU 虚拟化应用研究

目前国内外对 GPU 虚拟化研究的重点在虚拟化桌面应用、GPU 通用计算虚拟化和 GPU 虚拟化的安全问题等几个方面。

3.1 虚拟化桌面应用

虚拟化桌面应用指虚拟机在物理机上提供给用户透明的桌面使用环境,这些桌面使用环境通常具备良好的可迁移性。通常,多个被虚拟出的桌面环境要求完全隔离和平行,用户可以将多种不同的应用放在多个不同的虚拟桌面环境中完成。此外用户还能在各个虚拟桌面系统之间快速切换,同时根据各自的需求在各虚拟桌面环境中进行快速部署、用途设定和迁移。虚拟桌面技术需要提供良好的人机交互界面能力,因为大多数的桌面应用都需要进行频繁的用户交互。随着计算机的普及,大型 3D 游戏、视频播放等娱乐应用越来越流行,对画面显示能力的要求也

越来越高,高性能与流畅的图形显示能力已成为虚拟化桌面系统的必然要求。因此,对现有虚拟化技术中图形处理性能进行创新或改进成为研究主流,同时对现有的图形解决方案进行改进以提高虚拟化后图形的处理性能。

图形处理方式包括直接采用物理显卡和采用虚拟显卡即 GPU 虚拟化方式。其中,物理显卡的方式就是在对系统的资源进行整合与分配时,直接将物理显卡的控制权重新分配给特权的虚拟机。与真实物理主机相比,这种方式使特权虚拟机可以获得同样的图形效果,然而它与虚拟机动态分配、资源共享的设计理念不符,并且增加了使用成本;GPU 虚拟化方式即在虚拟机操作系统中模拟一块能够进行图形处理的物理显卡。目前系统级 GPU 虚拟化方式实现包括 Xen 虚拟帧缓存、VMware 虚拟图形显示处理器 GPU(Graphics Processing Unit)、虚拟网络计算机 VNC(Virtual Network Computing)以及独立于虚拟机管理器的图形加速系统^[11] VMGL (VMM-Independent Graphics Acceleration)。虚拟化桌面系统的图形加速方案通常采用 GPU 的 API 重定向虚拟化技术,研究成果有 GPU 通用计算框架 vCUDA^[12] (virtual VUDA)、GPU 图形虚拟化方案 VMGL 以及实现数据高速传输的域间通信系统。

3.2 GPU 通用计算虚拟化

目前针对 GPU 虚拟化的研究仍处于起步阶段,大多数都是针对 2D、3D 图形处理,在 GPU 通用计算方面涉及较少。随着 GPU 性能的提升,其优秀的通用计算能力越来越显著,与图形计算一样成为研究的热点。第一个 GPU 通用计算专用框架 CUDA 于 2007 年推出。vCUDA^[12] 采用在用户层拦截并重定向 CUDA API 的思路,在虚拟机中建立物理 GPU 的逻辑映像虚拟 GPU,实现了 GPU 资源细粒度划分、重组与再利用等功能,支持挂起/恢复、多机并发等虚拟机高级特性,使得在应用层实现 GPU 通用计算虚拟化成为可能。Pegasus 项目^[13] 则是基于 Xen 实现的虚拟机管理平台,为了提高 GPU 的资源利用率,它将 CPU 和 GPU 视为统一协同调度资源,解决了传统异构虚拟机环境中存在的 GPU 资源调度失配的问题。CheCUDA^[14] 则借鉴了 vCUDA 对 CUDA 状态进行管理和追踪的思路,讨论了在非虚拟化条件下实现 CUDA 状态恢复和保存的可能性,实现 CUDA 程序中的检查点方案。

3.3 GPU 虚拟化安全问题

在信息爆炸的大背景下,安全性成为 GPU 虚拟化不容忽视的问题,关于 GPU 虚拟化安全计算的研究同样处于起步阶段。

有关虚拟机的安全性研究包括:2010 年 Harnik^[15] 等人从隐蔽和侧面渠道等角度解决重复数据删除所带来的安全隐患;2012 年 Wu^[16] 等人断言:由于在物理内核上分配了虚拟内核,缓存隐蔽和侧通道是不实用的,他们提出了一种新的基于总线争用的隐蔽信道,通过使用原子操作来锁定共享内存总线。关于 GPU 特有的安全性研究有:

Maurice^[17] 等人在基于设备独占的 GPU 虚拟化上,分析了不同 VM 顺序访问同一 GPU 硬件可能存在的全局内存信息泄露问题,并结合现存的云计算平台研究了云环境下的 GPU 信息泄露问题。GPU 的全局内存清零只是纠错码的一个副作用,并没有涉及安全考虑;赵冰^[18] 提出了基于二进制哈希树的绑定方案,实现了多个 vPCR 与单一硬件 PCR 的绑定机制,建立起了真正意义上基于硬件基础的信任链,可大大提高 GPU 虚拟化平台对恶意软件的免疫力。

4 GPU 虚拟化解决方案

作为先进 GPU 技术与应用的提供者,NVIDIA 一直致力于通过将 GPU 强大的计算能力引入到通用计算领域,来实现并帮助企业“加速”业务体验和市场决策。目前,NVIDIA 主要包括 GeForce、Quadro、Tesla、Tegra、GRID 五条产品线^[19],可按照市场定位划分为面向消费级市场和面向企业级市场两大类,其中 GeForce 和 Tegra 分别定位在 PC 和移动市场,属消费类产品;Quadro、Tesla 以及 GRID 定位企业级市场,分别瞄准的是专业图形计算、高性能计算以及云计算市场。在企业级市场,NVIDIA 先是推出 Quadro 专业显卡来满足企业的专业设计需求,再通过推出 Tesla 主打高性能服务器市场,并根据企业“设计+仿真”一体化需求,推出了集图显与计算为一体的 MAXIMUS 产品,从而可以根据用户需求合理高效地分配计算资源,加速研发进程。市面上已有的 NVidia GPU 虚拟化解决方案包括 GPU pass-through、vSGA、GPU Sharing、VGX,表 2 为以思杰为代表的 GPU 虚拟化解决方案分类。

表 2 GPU 虚拟化解决方案分类

GPU 解决方案	优点	不足	适用人群
XenApp for GPU sharing	1. 对于服务器资源没有明确要求,虚拟机、物理机皆可 2. 应用虚拟化方式发布、部署、维护简单 3. 支持 DirectX 11, OpenGL 4.2 标准 4. 服务器利用率高,一台 XenApp 服务器可以支持 6~15 个并发用户使用 2D/3D 应用程序	在特定场景,特别是渲染大纹理或复杂场景时,会有锯齿出现,效果不理想	轻、中度图形工作人员和对渲染效果要求不高的场景
XenServer for vGPU(Beta)	1. 通过 API 调用 NVIDIA 的专业显卡,几乎能将显卡的全部性能发挥出来 2. 支持 DirectX 11, OpenGL 4.3 标准 3. NVIDIA 给每种虚拟出来的 GPU 都有对应的 GPU 型号,所使用与物理显卡相同的 NVIDIA 专用 Windows 驱动	1. XenServer 理论上已经完成了对 vGPU 功能的支持。但考虑到对客户负责的态度,暂时只有 Beta 版本 2. 只支持 XenServer6.2 版本	中度图形工作人员
GPU pass-through	性能最好,对于 OpenGL、CUDA、DirectX 各种协议支持良好	总价高,每台服务器最多插入 4 块物理显卡	重度图形工作人员

长远来看,GPU 虚拟化将充分释放 GPU 的并行计算

能力,大幅降低企业面对诸如大数据分析、动漫设计、电影特效渲染等高性能计算需求成本,同时提升云计算应用的服务质量。

5 结语

本文对 GPU 虚拟化技术进行了研究与分析,介绍了 GPU 虚拟化的主要方法;对虚拟化桌面应用、GPU 通用计算虚拟化和 GPU 虚拟化的安全问题进行了分析探讨,给出了相关研究的主要需求点和目标;对 GPU 虚拟化市场应用进行了梳理,给出了可行的解决方案。

参考文献:

- [1] 白洪涛. 基于 GPU 的高星级并行算法研究[D]. 长春: 吉林大学, 2010.
- [2] 马秀芳, 李红岩. 计算机虚拟化技术浅析[J]. 电脑知识与技术, 2010, 16(33): 9408-9412.
- [3] VMWARE. VMware workstation[EB/OL]. www.vmware.com/products/ws/.
- [4] 石磊, 邹德清, 金海. Xen 虚拟化技术[M]. 武汉: 华中科技大学出版社, 2009.
- [5] XEN. Xen VGA passthrough[EB/OL]. wiki.xensource.com/xen-wiki/XenVGAPassthrough.
- [6] INTEL. Intel VT-d [EB/OL]. www.intel.com/technology/itj/2006/v10i3/2-io/1-abstract.htm.
- [7] BELLARD F. QEMU, a fast and portable dynamic translator[C]. In: Proc of the annual conference on USENIX Annual Technical Conference, Berkeley, 2005: 41-41.
- [8] VMWARE. VMware SVGA device developer kit[EB/OL]. vmware-svga.sourceforge.net.
- [9] HUMPHREYS G, HOUSTON M, NG R, et al. Chromium: a streamprocessing framework for interactive rendering on clusters [C]. In: Proc of 29th Annual Conference on Computer Graphics and Interactive Techniques. New York, 2002: 693-702.
- [10] HANSEN J G. Blink: 3d display multiplexing for virtualized applications[R]. Technical Report, ISSN: 0107-8283, 2006.
- [11] LAGAR CAVILLA H A, TOLIA N, SATYANARAYANAN M, et al. VMM-independent graphics acceleration[C]. In: Proc of Virtual execution environments, New York, 2007: 33-43.
- [12] SHI L, CHEN H, SUN J. vCUDA: GPU accelerated high performance computing in virtual machines[C]. In: Proc of International Parallel & Distributed Processing Symposium. Rome, 2009, 1-11.
- [13] GUPTA V, SCHWAN K, TOLIA N, et al. Pegasus: coordinated scheduling for virtualized accelerator-based systems[C]. In Proc of USENIX Annual Technical Conference, Berkeley, 2011: 3-5.
- [14] TAKIZAWA H, SATO K, KOMATSU K, et al. CheCUDA: a checkpoint/restart tool for CUDA applications[C]. In: Proc of International Conference on Parallel and Distributed Computing Applications and Technologies. Higashi Hiroshima, 2009: 408-413.
- [15] HARNIK D, PINKAS B, SHULMAN-PELEG A. Side channels in cloud services: reduplication in cloud storage[J]. Security & Privacy, 2010, 8(6): 40-47.
- [16] WU Z, XU Z, WANG H. Whispers in the hyper-space: high-speed covert channel attacks in the cloud[C]. In Proceedings of the 21st USENIX conference on Security symposium, Berkeley, CA, USA, 2012: 9-10.
- [17] MAURIVE C, NEUMANN C, HEEN O, et al. Confidentiality issues on a GPU in a Virtualized Environment[C]. In Proceedings of the 18th International Conference on Financial Cryptography and Data Security (FC), Barbados, BARBADOS, 2014.
- [18] 赵冰. GPU 虚拟化中安全问题的研究[D]. 成都: 电子科技大学, 2014.
- [19] 李翔, 熊东旭. 化云为雨. NVIDIA“加速”云落地[EB/OL]. http://www.e-works.net.cn/interview/leader_597.htm.

(责任编辑: 杜能钢)

Applications of New Trends on GPU Virtualization Techniques

Abstract: GPU, as a type of very high cost efficient computing resource, has become an important component to construct high performance computing system. Meanwhile, GPU virtualization also has been paying more and more attention so as to improve practical usage efficiency. First, a brief introduction on GPU virtualization techniques is given in this paper, and three main GPU virtualization techniques including exclusive use, GPU virtual equipment simulation and application layer interface virtualization are detailedly described. Second, virtual desktop application, GPU virtualization for general computation, and the security issues of virtual GPUs are analyzed. At last, several GPU virtualization solutions for some typical applications are concluded out.

Key Words: GPU Virtualization; Virtual Desktop; General Computing Virtualization; Application Analysis