Virtual Machine Monitors: Current Technology and Future Trends阅读报告

文章发表在2005年IEEE Computer,vol38,issue 5，作者是来自VMware和斯坦福的研究者。 虚拟机技术最初是用来解决大型机（mainframe computing problems）问题的。最近重新再商用平台上出现，用来解决安全，可靠和管理的问题。

文章内容

在1960年末，虚拟机技术已经出现。软件抽象层。人们发现这是一种充分利用大型机这种珍稀计算资源的方法。VMM技术热潮在1980年到1990年结束，现代多任务OS的出现和硬件的成本下降。大型机->微型机->PC。计算机体系结构和硬件不对VMM提供必要的支持。 2005年左右，虚拟机技术又火起来。 VMM技术的衰落与兴起之间发生了什么？ 在19世纪90年代，出现了由硬件和操作系统导致的问题：MPP(massively parallel processing)机器不能够运行多个OS，难以编程。通过虚拟机技术可以使这些机器变成现有的OS。这个项目的参与者成为了Vmware的初始团队。vmm技术在商用平台上的使用引发了研究兴趣。

为何重新兴起

硬件低廉和现代OS的出现，VM退场，出现新问题：较多的硬件的管理负载和增加的功能使得OS脆弱->系统和应用老崩溃->每个机器上运行一个应用->对硬件的需求提升，重大的管理开销->虚拟机技术提高硬件使用效率和减少管理开销。 VMM更像是安全和可靠性的解决方案。像迁移和安全这种情况不适合在现代OS中实现，反而适合在VMM层中实现。

将硬件和软件解耦合

硬件和客户OS之间的中间层，控制客户OS如何访问硬件资源。使得来自不同的制造商的机器（不同的IO）看起来是相同的。 VMM对虚拟机的软件状态提供完全的封装。 映射虚拟机资源到硬件，重新映射，迁移，负载均衡，硬件失败，扩展性。管理员可以对虚拟机进行任意的suspend和resume或者checkpoint和roll back. OS也具备了可移动性，可以将虚拟机镜像拷贝到可移动设备。 VMM：虚拟机之间的隔离；虚拟机复用同一硬件平台；一个应用导致的OS崩溃不会殃及整个系统。

VMM实现时的问题

对软件来说透明，同时对软件访问硬件时的操作有控制，可以在其中加入操作。 设计目标：兼容性：运行legacy软件；性能：软件与运行在物理机上体验相同。简单性：VMM的错误才不会导致所有的VM崩溃。

CPU虚拟化

一个CPU架构是可虚拟化的，如果其支持VMM的基本技术：直接执行---在真实机器上执行虚拟机，同时

VMM拥有对CPU的最终控制。 实现基本的直接执行：在CPU的非特权态运行虚拟机的特权和非特权代码，而VMM运行在特权状态。当虚拟机执行特权操作时，CPU陷入到VMM。 提供可虚拟化的CPU结构的关键是：提供trap语义，使得VMM能够安全，透明，直接地使用CPU资源来执行虚拟机。

挑战：大多数现在的CPU设计不支持虚拟化，包括X86。另外一个挑战是：非特权指令让CPU访问特权状态。

技术：有几项技术解决了如何在不支持虚拟化的CPU上实现VMM。最流行的：paravirtualization，结合快速二进制翻译的直接执行。 Disco,用于不支持虚拟化的MIPS架构，使用了paravirtualization.该设计将MIPS的中断flag变为简单的一个虚拟机中的内存定位而不是处理器中的特权寄存器。 para虚拟化的最大缺点是：不兼容性。运行在para虚拟化的VMM上的任何OS都必须移植到该体系结构上，需要OS制造商的合作。 结合快速二进制翻译的直接执行技术是vmware的。还有的解决的技术是：开发二进制翻译器，在不同的指令集的CPU之间翻译代码。VMM的基本技术就是在二进制翻译器的控制下运行特权模式的代码。二进制翻译负载小，启用trace cache的话，速度较快。同时还能优化直接执行，消除了很多trap. 未来支持：intel:Vanderpool和AMD的Pacifica技术。支持VMM。直接支持直接执行的CPU将成为可能。硬件支持会使得性能大幅提升。

内存虚拟化

传统的技术：使VMM维护一个虚拟机内存管理的数据结构的shadow。shadow page table VMM可以动态地控制各个虚拟机需要多少内存，根据其对应的需求。

挑战：VMM的虚拟内存子系统必须周期性地重新声明一些内存，通过page一部分的虚拟机到磁盘。然而，客户OS对于哪些页是好的page out的候选，其有更好的信息。为了解决这一问题，vmware采取了类似paravirtualization的方法：一个客户机内运行的balloon进程可以与VMM通信。 第二个挑战是：现代OS和应用的大小。运行多个虚拟机或浪费大量的内存--多个虚拟机之间存储相同的代码和数据。解决：基于内容的页分享。VMM追踪物理页的内容，观察其是否相同。若相同，修改shadow页表。如果内容后面变化了，VMM给每个虚拟机自己的页拷贝。

未来的支持：OS对页表做频繁的更改，使影子拷贝维持当前最新会造成很大的开销。硬件管理的影子页表一直出现在大型机的虚拟化架构中。这可能是x86的一个方向。资源管理还是未来研究的重点。

IO虚拟化

IBM的大型机的IO子系统采用的是channel-based的架构。对IO设备的访问都是通过独立的channel处理器进行。通过使用channel处理器，VMM可以安全地将IO设备的访问导出到虚拟机。相比于使用trap到VMM，这样虚拟机可以直接访问设备。 挑战：当前的计算环境以及各种丰富的IO设备，使得IO的虚拟化很难。写一个可以与各种设备talk的VMM层很难。现代的一些网络和图像相关的外设对性能要求很高，负载要小。导出一个标准的设备接口意味着虚拟化层必须能够和计算机的IO设备进行通信。VMware workstation的虚拟层使用宿主机器的驱动。就是IO的VMM在主机OS之上，但是其他的VMM与主机OS是同层的。 这种hosted的架构的好处：安装vmware就像安装应用一样；可以容纳各种丰富的IO设备；VMM可以借助宿主环境提供的调度，资源管理和其他服务。 缺点：（向x86服务器市场推进时出现问题）：增大了IO设备虚拟化的负载。每个IO的请求都必须要传给宿主OS才可以。对于高性能的网络和磁盘子系统，这种负载不能接受；现代OS不提供对虚拟机的性能隔离和服务保证。 ESX服务器采取了更传统的方法：VMM直接运行在硬件上，无需宿主

OS。ESX对高性能的网络和磁盘支持好。ESX服务器内核使用Linux内核的设备驱动器直接与设备talk，极大降低了设备的负载。 另外的性能优化：导出特殊优化过的虚拟IO设备。这种方法需要客户OS使用特殊设备驱动来访问IO设备。

未来的支持：像CPU，趋向于对IO的虚拟化进行硬件支持。有了硬件的支持，将这些channel IO设备直接传到虚拟机中的软件是可行的。无需陷入到VMM中。 实现DMS的IO设备需要地址重映射。同时为了要保持隔离的属性，设备应该能够只访问属于该虚拟机的内存。 在一个使用相同设备但是有多个虚拟机的系统中，VMM需要一个高效的机制来保证路由完整设备中断到合适的虚拟机中。最后，虚拟化的IO设备需要接口到VMM来确保硬件和软件之间的隔离，同时确保VMM可以迁移和对虚拟机做checkpoint.

未来

VMM的未来以及业界对虚拟化技术的需求

服务器侧

在数据中心中管理员能够通过一个控制台来控制上百台物理机上上千台虚拟机。不需要逐一对这些机器进行配置，模板和管理策略。例子是vmware的Virtual Center。这种虚拟机到硬件资源的映射是高度动态的，热迁移能力，如VMware’s VMotion技术。可以根据数据中心的需求将虚拟机在物理机之间快速迁移。现在主要是人工进行迁移，将来应该是自动迁移，创建和销毁。

机房之外

虚拟化技术从机房到桌面，虚拟机提供了有效统一的方法来重塑桌面管理。解决VMM层的问题可以对所有运行在虚拟机内的软件有好处。OS无关性减少了购买和维护基础设施的必要。改变了用户对计算机的认知。虚拟机增加的移动性改变机器的使用。将用户的计算环境在局域和广域进行迁移是可行的，这是通过项目验证的。USB。虚拟机基于的环境的增加的动态性将需要更多的动态网络拓扑，虚拟交换机，虚拟防火墙以及overlay network.逻辑计算环境与物理地点解耦合。

安全提升

当前的OS隔离性不好；相关研究：使用VMM作为高级入侵检测。以及使用VMM层来分析攻击者可能造成的危害。这些系统不仅获得了较好的攻击抵抗能力（在VM外进行操作），也从在硬件层次插入和监控虚拟机内的系统中获益。虚拟化技术可以限制虚拟机的访问网络，在访问前检查虚拟机。虚拟机变成了一个有效的工具在限制恶意软件传播方面。 适合构建高保证的系统。high-assurance system。 VMM支持在不同的安全等级运行多个软件栈。 适合构建可信计算环境。VMM可以向远程参与方认证虚拟机内运行的软件。不同安全内容使用不同的虚拟机访问，向远程的服务器提供身份证明。 灵活的资源管理应对DDOS攻击。

软件分布

软件公司可以在发布软件时，发布整个虚拟机，包含复杂的软件环境。简化了用户安装和配置。

想法

VMM的复兴改变了软件和硬件设计者的认知。为部署新的OS提供了向后的兼容性。VMM在未来计算的发展中有重要的作用。 结合网络的研究：动态拓扑，虚拟交换机和虚拟防火墙，NFV以及云计算环境下的网络攻击。