嵌入式领域中的虚拟化技术

潘禹辰

浙江大学软件学院，浙江 宁波 315103

E-mail ：vincent.d.pan@gmail.com

摘　要：介绍了虚拟化技术的现状，从层次结构和技术特点两个角度论述了虚拟化技术的分类，并介绍了几种主流虚拟化技术的实例，之后详细阐述了将虚拟化用于嵌入式领域的原因，并结合 VirtualLogix 公司的 VirtualLogix VLX 分析了在嵌入式领域中虚拟化技术的具体实现和应用前景。

关键词：虚拟化；嵌入式；VMM；虚拟机（VM）

1　概论

近年来,虚拟化技术无论是在学术界还是企业界,都刮起了不小的旋风,并且势头不减。Gartner 两位分析师 Carl Claunch 和 Dave Cearley 在 2008 年的 Symposium ITxpo 大会上列出了未来 3 年最值得关注的十大关键技术，其中虚拟化技术位居榜首。虚拟化是指计算机软件在一个虚拟的平台上而不是真实的基础上运行。虚拟化技术可以扩大硬件的容量，简化软件的重新配置过程。其中 CPU 的虚拟化可以单 CPU 模拟多 CPU 并行运行，允许一个平台同时运行多个操作系统，并且应用程序都可以在相互独立的空间内运行而互不影响，从而显著提高计算机的工作效率和安全性，这也是越来越多的人对虚拟化产生兴趣的一个原因。数据表明，虚拟化给企业带来的效果是显著的。戴尔 IT 战略专家 Matt Brooks 坦言：“与部署和运行物理服务器相比，在 3 年的时间里，戴尔部署的每台非生产用虚拟机节省了约 6,500 美元，每台生产用虚拟机节省了约 9,300 美元。自实施虚拟化以来，戴尔已经将应用软件的部署时间从平均 45 天缩短到了仅需 4 天——缩短了 90%……”但是到目前为止，虚拟化技术还大多用于服务器领域，即企业计算。作者试图通过本文在介绍虚拟化技术的基础上，探讨将虚拟化技术用于嵌入式系统领域的效果和前景，并用实例来加以说明。

2　虚拟化技术介绍

虚拟化并不是新兴技术，它的历史甚至能追溯到 20 世纪 60 年代。最早的包括 IBM 7044、MIT 在 IBM 704 上开发的 CTSS（Compatible Time Sharing System）以及曼彻斯特大学的 Atlas 项目，这些都是请求页面调度和监管进程调用的先驱。虽然如此，虚拟化技术也曾经因各种原因销声匿迹，但随着信息技术的发展和需求的驱动，人们对虚拟化的热情又高涨起来。

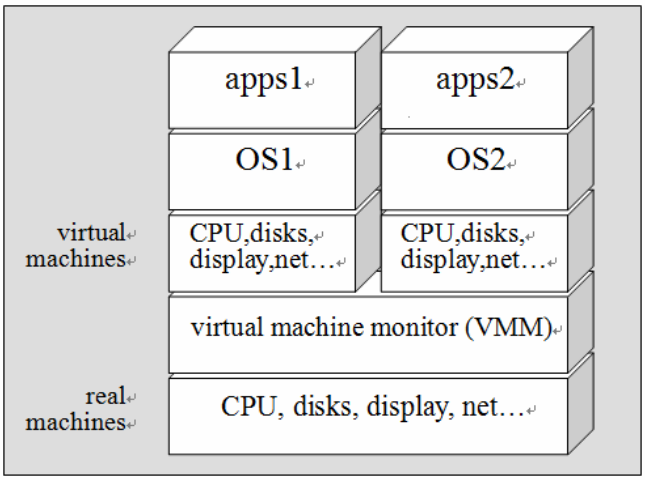

图 1：硬件层虚拟化

2.1　虚拟化技术的层次分类

现代计算机系统是有层次结构的：从最下层的硬件层，到中间的系统软件层，再到最上层的应用软件层。虚拟化技术根据实现于系统的不同层次可以分为以下类：

1. 硬件层虚拟化。硬件层虚拟化实现的虚拟机直接位于硬件之上，运行于虚拟机之上的软件就如同运行于正真的硬件之上，其结构如图 1 所示。

2. 操作系统层虚拟化。这种虚拟化技术所实现的虚拟机位于操作系统和应用软件之间。运行于虚拟机上的应用软件都是针对某一种特定的操作系统开发的，FreeBSD Jails 就是这种技术的代表。

3. 高级语言虚拟化。这种技术所实现的虚拟机直接作为一个应用软件运行于操作系统之上。它主要是运行一些针对一定的抽象定义所编译过的程序。例如现在非常盛行的 Java 技术就是以这种技术为基础。

2.2 虚拟化技术的技术分类

上面我们介绍了根据虚拟机位于系统的层次而分的不同种类。而实际上，从实现技术的角度，我们通常把虚拟化技术分为硬件仿真、完全虚拟化和超虚拟化。

1. 硬件仿真

它是指在宿主机上建立一个硬件虚拟机来仿真目标硬件，如图 2 所示。他可以说是虚拟化中实现技术最复杂的一种。因为它要对在虚拟机上运行的每一条指令都要在宿主硬件上进行仿真，所以它的执行效率非常低。对于一些高保真的仿真，其速度一般会比直接在目标硬件上执行要慢成百上千倍。但是它还是我们在通常的工程实践中经常用到的技术。例如嵌入式系统工程师会在一些仿真环境中（如：Bochs, Qemu 和 skyeye）进行系统开发。

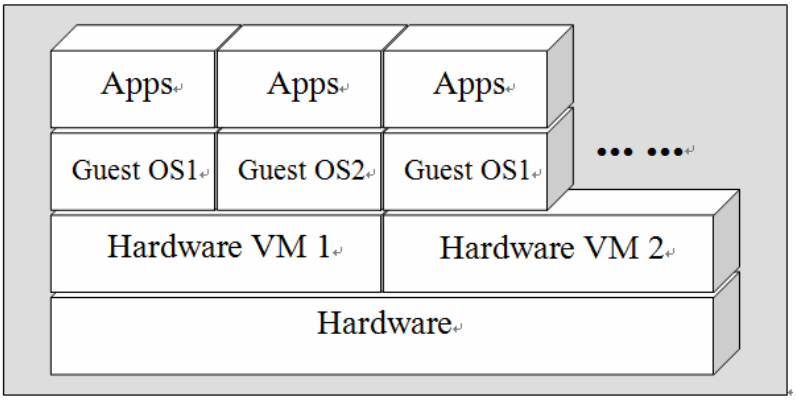

图 2：硬件仿真

2. 完全虚拟化（Full virtualization）

这种虚拟化技术是加入了一个虚拟机监视器（Virtual Machine Monitor, VMM）—— Hypervisor，它运行在底层硬件之上。Hypervisor 主要负责为硬件和客户操作系统（Guest OS）提供协调。其中最主要的职责就是将一些客户操作系统要执行的特权指令捕获下来，然后由它在硬件上进行执行，如图 3 所示。

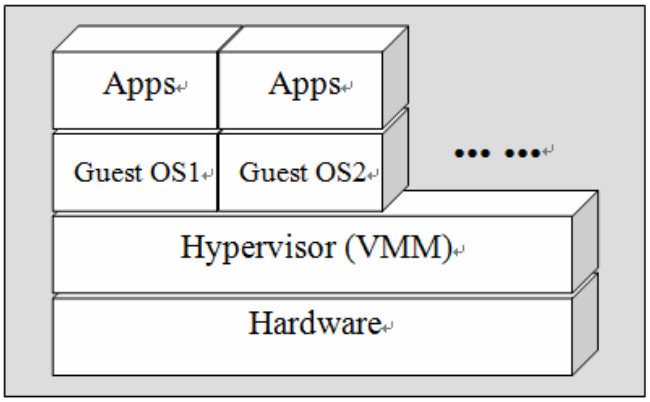

图 3：完全虚拟化

完全虚拟化最大的特点就是客户操作系统不用修改就能运行于 VMM 之上，因此它具有很好的兼容性和同时支持异种 OS 或不同版本 OS 的能力。正是由于这个优点，完全虚拟化技术在开始就受到客户的青睐，而在完全虚拟化技术上做得最成功的就属 VMware 了。

1998 年 1 月，Stanford 大学的 Mendel Rosenblum 教授和他的几个学生成立了 VMware 公司。1999 年 VMware 推出了基于 X86 的完全虚拟化的系统虚拟机即 VMware Workstation，一款基于主机模型的虚拟机。所谓主机模型，即 VMM 运行在主机 OS 上(Host OS)，这使得 VMM 可以充分利用主机 OS 所提供的设备驱动及底层服务而不需要去除原来机器

上已经安装的操作系统，但也正是由于主机 OS 的“掺和”使得系统性能损失不少。在 2001 年 VMware 又推出了 VMware ESX Server 和 VMware GSX Server 以满足服务器市场的需求。

3. 超虚拟化（paravirtualization）

超虚拟化是最近几年最炙手可热的虚拟化技术，它也是通过一个 hypervisor 来实现多个客户操作系统对底层硬件的共享访问，另外还同时将与虚拟化有关的代码集成到了操作系统本身中，如图 4 所示。相对于完全虚拟化，这种方法最大的特点是不再需要捕获特权指令，但是需要为 hypervisor 修改客户操作系统。

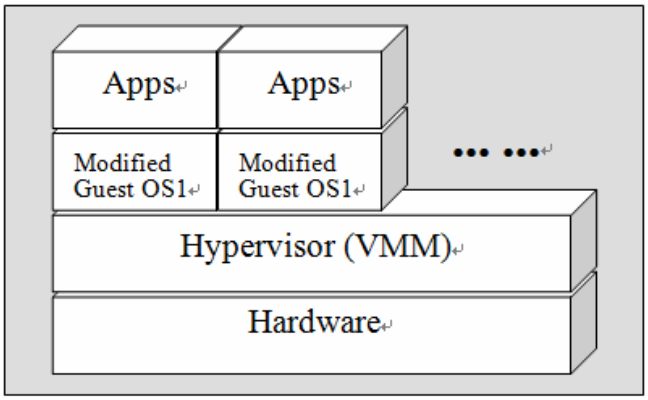

图 4：超虚拟化

由上述特点可知，虽然这种虚拟化技术会给技术人员带来额外的工作量，但是通过对内核的修改能显著地提高虚拟系统的性能，并且运行于客户操作系统这也是它在现在备受推崇的原因。目前在超虚拟化上做得最成功的当属从社区里走出的 Xen。

Xen 最早是在剑桥大学作为一个 Linux 研究项目开发的，之后在 Linux 社区的推动下逐步发展壮大，并且吸引了 intel 等大公司加入其中。Xen 是一个基于开源(Open Source)的混合模型系统虚拟机。Xen 将客户机操作系统称之为虚拟域(Domain)，其中 Domain 0 为服务域，拥有绝大部分的 I/O 资源(其他部分为 VMM 所有)并向其他虚拟域提供设备模型以及控制平台。由于它是一种超虚拟化技术，所以只有那些修改过的操作系统才可以通过 Xen 进行虚拟化。正是这个原因，使之不能对一些闭源操作系统做到很好的支持，如：windows。但是它的高性能以及对 Plan 9、NetBSD、FreeBSD 和 Solaris 的广泛支持也为它赢得了很多客户的青睐。

3 嵌入式领域中的虚拟化

上面基本介绍了现在的各种主流的虚拟化技术，但它们几乎都还是主要应用于企业计算或者叫服务器领域。当今世界和虚拟化同样繁荣的还有一个领域，那就是嵌入式系统。不管是 Intel、Motorola 等硬件厂商，还是微软、Google 等软件巨头，都将注意力移到了嵌入式领域。同时嵌入式技术本身随着发展也逐渐碰到了一些问题，而虚拟化则被认为是解决当前嵌入式领域所面临的一些问题的有效手段。

3.1 嵌入式系统的挑战

嵌入式系统发展到今天，人们对它的理解不再是以前功能简单、界面粗糙的“小玩意儿”。现在的高端嵌入式系统（如：智能手机、PDA 等）不管是在功能上还是在性能上甚至能和 PC 机媲美。现在的大多数嵌入式产品也不再像以前，一生产出来里面的软件就不能再被修改和更换，直至系统报废。但是随着嵌入式技术的发展，客户的要求也不断提高，他们日益增长的对嵌入式产品功能和性能的需求和嵌入式技术不能满足其需求构成了一个让全世界工程师困扰的问题。例如用户往往想要用到为不同操作系统开发的应用软件。一个典型的例子就是一位在硅谷工作的经理总是随身带着三部手机：他用 RIM 公司的黑莓 Curve 8300 来收发电子邮件，用摩托罗拉的 Razr 手机来打电话，用苹果 iPhone 来上网，他显然对这样的结果很不满意。有没有方法让他带一部手机就实现他的需求，我们现在可以提供一种答案：虚拟化。

3.2 嵌入式虚拟化

将虚拟化引入到嵌入式系统最大的好处就是能使之实现在同一硬件平台上运行多个操作系统。这一效果能连带解决在嵌入式领域面临的多个问题。首先是开发周期的问题。当下，程序员如果要为每一种嵌入式操作系统重新编写应用软件，整个开发周期可能要持续几个月，这对企业来说是一个极大的浪费。而虚拟化技术可以让程序员直接将已有的应用软件移植到目标机上。例如为了满足上面那位经理的需要，程序员可以把浏览器、电子邮件软件、电话功能等分布在三种嵌入式操作系统上的软件集成到同一部手机上，这将为企业节省很多的时间。其次，随着越来越多的嵌入式产品能够下载或运行第三方应用软件，安全因素变得尤为重要。运用虚拟化技术，嵌入式产品厂商可以运行一种高性能的嵌入式操作系统来控制重要的基本功能（如一些实时响应），同时在另一个功能强大的嵌入式操作系统上（如 Linux、WinCE 等）运行满足用户其他需求的应用软件。这样发生了安全问题，也能保证基本的功能和一些重要数据不受到那些其他应用软件的影响。另外，从成本角度来

说，虚拟化技术还可以减少嵌入式系统工作所需的芯片数量。例如上述情况就是用一个或者两个芯片就实现了三个以上芯片的功能。这样，即使要为产品换用更高性能的芯片，企业也能从中节约大量成本。

4 嵌入式虚拟化应用实例

众所周知，嵌入式系统的一个重要的特点就是资源有限，不管是 CPU 还是存储空间。而现在的嵌入式工程师主要的一项工作就是怎么才能提高对资源的利用率。嵌入式系统工程师听到要将虚拟化引入嵌入式系统时，首先考虑的问题就是拿出一部分资源给虚拟化划不划算。其实，现在已经有一些先行者通过成功的嵌入式虚拟化解决方案打消工程师们的这个疑虑。

在已有的嵌入式虚拟化解决方案中，VirtualLogix 公司的 VirtualLogix VLX 是其中的佼佼者。VirtualLogix VLX 是典型的嵌入式虚拟化技术，它能实现在保持嵌入式系统实时性的基础上同时在同一处理器上运行多个操作系统作为客户操作系统。具体地说，VirtualLogix VLX 实际上就是在硬件层和系统软件层之间的一个薄抽象层（如图 5 所示）。这样，硬件资源就能在客户操作系统对硬件资源（如：CPU、real-time clock 等）提出请求时，通过虚拟化硬件来满足它们的需求。

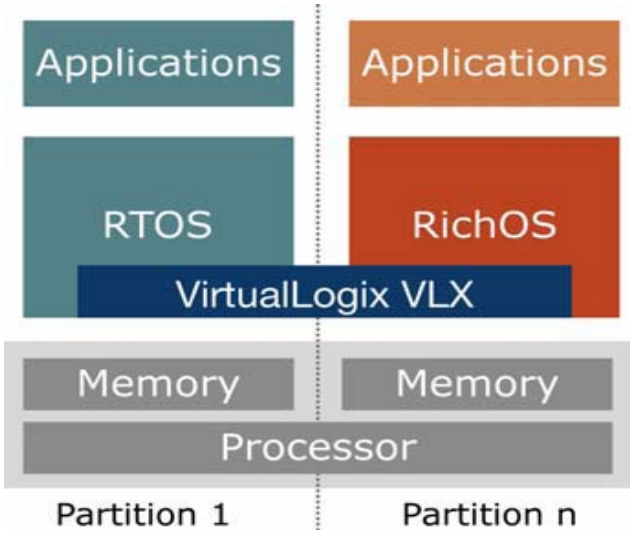

图 5[3]：VirtualLogix VLX

VirtualLogix VLX 的一个技术亮点就是引入了分区（Partition）的概念。存储器经过分区后分配给各客户操作系统，这样做的好处在于：各个客户操作系统可以运用自己的内存管理机制而不必打扰其他操作系统。这也就为有 MMU 的操作系统（Linux 和 WinCE 等）和没有 MMU 的操作系统（Nucleus 和 VRTx 等）和谐地运行于同一处理器上。

同时我们看到，这种技术实现了我们上面所讲的需求：在保持其他应用功能的基础上，通过实时操作系统（RTOS）增强了嵌入式系统的性能。同时由于虚拟层的引入增强了系统的安全性，如图 6 所示。对于一些通讯设备，通信协议栈是至关

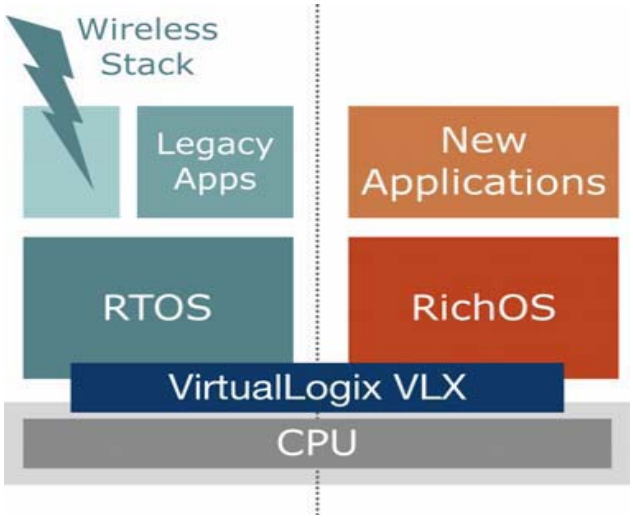

图 6[3]：入侵防范

重要的，如果有一些人入侵到其中，他完全可以通过破坏协议栈的方式干扰整个局部网络。一个极端的例子就是，一部蜂窝电话（cell phone）因协议栈被入侵而沦为一部干扰发射台，从而影响整个 cell。如果按传统的想法，通过加密技术来解决，那庞大的代码量是让人头痛的。虚拟化使得以前运行于特权模式（privileged mode）的 RTOS，现在运行于 de-privileged 模式，这就有效阻碍了入侵对硬件和其他客户操作系统的破坏。前不久，TI 的 C6474 产品的推出标志着运用虚拟化技术的嵌

入式产品从实验室走到了市场。这个系列的 DSP 利用 VLX 技术，使得以前需要两个专用 DSP 与一个通用处理器才可支持的各种任务，只需单个 DSP 便可高效实现，同时不需重组硬件，就能通过 DSP/BIOS 内核与 Linux 软件构建不同的产品，并且对功能和安全性都有显著提升。

5 结论

综上所述，虚拟化正在从很多方面吸引着嵌入式。但是我们应看到虚拟化本身还没有发展得十分成熟，那些主要用于服务器领域的各种主流虚拟化技术并不能直接引入嵌入式领域，而要根据嵌入式系统的特点进行适当的修改。但可以肯定的是虚拟化是大势所趋，将虚拟化引入但嵌入式领域也是未来嵌入式发展的一个方向。

References:

[1] Gernot Heiser. The role of virtualization in embedded systems [C]. First Workshop on Isolation and Integration in Embedded Systems(IIES'08) April 1, 2008, Glasgow, UK.

[2] Mendel Rosenblum. The Reincarnation of Virtual Machines [J]. Queue, 2004, Volume 2, Issue 5.

[3] VirtualLogix. Real-Time Virtualization White Paper. http://www.virtuallogix.com. 2006

[4] M. Tim Jones. Virtual Linux:An overview of virtualization methods, architectures, and implementations. http://www.ibm.com/developerworks/linux/library/l-linuxvirt/?S_TACT=105AGX52&S_CMP=cn-a-l.

[5] DONG Yaozu, ZHOU Zhengwei. X86-based System Virtual Machine Development and Application [J]. Computer Engineering, July 2006, 71-73.

[6]David Chisnall. Title [M]. The Definitive Guide to the Xen Hypervisor: Prentice Hall, 2008.