国际信息工程先进技术译丛

WILEY

虚拟网络——下一代互联网的多元化方法

Virtual Networks: Pluralistic Approach for the Next Generation of Internet

[巴西] Otto Carlos M.B. Duarte
[法国] Guy Pujolle 编著

冯玉芬 母景琴 王玲芳 等译

机械工业出版社
CHINA MACHINE PRESS

国际信息工程先进技术译丛

虚拟网络——下一代互联网的多元化方法

[巴西] Otto Carlos M. B. Duarte
[法国] Guy Pujolle 编著
冯玉芬 母景琴 王玲芳 等译

机械工业出版社

图书在版编目（CIP）数据

虚拟网络：下一代互联网的多元化方法/（巴西）杜拉特（Duarte, O. C. M. B.），（法国）普杰（Pujolle, G.）编著；冯玉芬等译. —北京：机械工业出版社，2014.12

（国际信息工程先进技术译丛）

书名原文：Virtual Networks: Pluralistic Approach for the Next Generation of Internet

ISBN 978-7-111-48726-5

Ⅰ.①虚… Ⅱ.①杜…②普…③冯… Ⅲ.①虚拟网络 Ⅳ.①TP393

中国版本图书馆 CIP 数据核字（2014）第 282689 号

机械工业出版社（北京市百万庄大街 22 号　邮政编码 100037）

策划编辑：张俊红　责任编辑：张俊红

版式设计：霍永明　责任校对：刘怡丹

封面设计：马精明　责任印制：乔　宇

北京机工印刷厂印刷（三河市南杨庄国丰装订厂装订）

2015 年 1 月第 1 版第 1 次印刷

169mm × 239mm · 13.5 印张 · 267 千字

0 001—2 000 册

标准书号：ISBN 978-7-111-48726-5

定价：69.80 元

凡购本书，如有缺页、倒页、脱页，由本社发行部调换

电话服务　　网络服务

服务咨询热线：(010)88361066　机 工 官 网：www. cmpbook. com

读者购书热线：(010)68326294　机 工 官 博：weibo. com/cmp1952

(010)88379203　教育服务网：www. cmpedu. com

封面无防伪标均为盗版　　金　书　网：www. golden-book. com

作为一本专门讲述虚拟网络的科技专著，本书内容涵盖虚拟化技术、两种虚拟化平台及其管理接口，综述了虚拟化联网的现有控制算法，同时描述了使用 Xen 作为虚拟化工具进行报文转发的主要挑战，并详细描述了虚拟网络局部控制的一个建议方案。

本书适合于计算机网络/通信领域的高年级本科生、研究生和研究人员，尤其适合对未来互联网感兴趣的读者。

译 者 序

在不久的将来，互联网将发生巨大变化，为迎接这种变化，信息领域的从业者需要及早介入。虚拟网络作为构建未来互联网的核心技术，在国内外被炒得如火如荼。目前国内在863计划、973计划以及自然科学基金方面对未来互联网都有项目支持，但国外对中国未来互联网的评价是处于摇篮阶段，因此国内急需这方面的内容全面深入的图书和资料。在此背景下，我们推荐引入本书。

就技术的前瞻性而言，本书描述一种多元论的方法，作为后互联网协议（IP）环境的一种新架构，这是未来互联网界的普遍共识之一。本书中给出的多数试验结果来自Horizon项目，这是由法国ANR（Agence Nationale de la Recherche）和巴西Finep（Financiadora de Estudos e Projetos）资助的一个两国共同参与的研究项目。他山之石可以攻玉，希望借此促进国内未来互联网的技术发展。

本书内容涵盖虚拟化技术、两种虚拟化平台及其管理接口，综述了虚拟化联网的现有控制算法，同时描述了使用Xen作为虚拟化工具进行报文转发的主要挑战，并详细描述了虚拟网络局部控制的一个建议方案。本书结构如下：

第1章讨论虚拟化技术，描述Xen、VMware和OpenVZ虚拟化的主要特征并识别出它们的性能折中；第2章详细描述Xen和OpenFlow虚拟化平台，并给出二者的性能分析；第3章讨论在前一章讨论的两种平台的管理工具；第4章描述语境感知的技术和多智能体系统；第5章讨论虚拟化联网的现有控制算法，也分析了使用Xen作为虚拟化工具进行报文转发的主要挑战，并详细描述了虚拟网络局部控制的一种提案；第6章描述引导系统，给出一个多智能体自管理原型；第7章讨论管理和控制功能；第8章详细地描述了用于系统架构的虚拟化技术。

本书由王玲芳负责第1~3章的翻译、全书统稿和校对工作，冯玉芬负责第4~6章的翻译工作，母景琴负责第7~8章的翻译工作。本书在翻译过程中，李虹、潘东升、李冬梅、吴秋义、王弟英、吴璟、游庆珍、李传经、王领弟、王建平等同志参加了部分的翻译工作，在此表示感谢。同时感谢机械工业出版社，感谢出版社的编辑和相关同志。

需要指出的是，本书的内容仅代表原作者个人的观点和见解，并不代表译者及其所在单位的观点。另外，由于翻译时间比较仓促，疏漏、错误之处在所难免，敬请读者原谅和指正。

译者

2015年初于北京

原书前言

当前，研究共同体存在大量积极的工作来重新思考互联网架构，应对互联网当前的限制并支持新的需求。许多研究人员得出结论，对于所有用户和网络提供商的需求而言，不存在均码（one-size-fits-all）的解决方案，因此倡议一种多元论的网络架构。这种新的架构彻底改变了互联网，因为它允许不同协议栈的共存，在同一物理基层上同时运行。因此，本书描述一种多元论的方法，作为后互联网协议（IP）环境的一种新架构。这种后IP架构主要基于带有一个引导系统的虚拟网络，它能够处理各种约束。这种引导系统是面向智能的，并有助于选择最佳参数，通过来自多智能体系统的机制，优化网络的行为。确实，面向自治的架构与网络设备的每部分（路由器、设备等）相关联，这是一种身临其境的方法，将被用来确定语境，并选择和优化控制算法和参数。

本书建议使用的后IP网络的另一个非常重要的概念是网络虚拟化，它将网络抽象为虚拟域（分片/基层）。一个虚拟域代表虚拟路由器而不是物理路由器实例的一个一致的功能组。在这个动态的多栈网络中，多个虚拟网络共存于一个共享的基层之上。这些域将使用引导系统来分配物理资源，并确定哪个虚拟网络将由一名客户使用。在这个语境中，一个服务提供商将能够同时运行具有不同性能和安全等级的多个端到端服务。必要时，可创建和删除虚拟网络。虚拟化支持网络的物理资源更好地加以使用，为客户带来适合的网络。

在本书中给出的多数试验结果来自Horizon项目，这是由法国ANR（Agence Nationale de la Recherche）和巴西Finep（Financiadora de Estudos e Projetos）资助的一个两国共同参与的研究项目。这个国际团体由5个学术和3个业界合作方组成。学术合作方是UPMC—巴黎第6（Laboratoire d'Informatique de Paris 6-LIP6）、Telecom SudParis、Universidade Federal do Rio de Janeiro（UFRJ）、Universidade Estadual de Campinas（Unicamp）和Pontifícia Universidade Católica do Rio de Janeiro（PUC-Rio）。业界合作方是Ginkgo-Networks SA（引导知识平面方面的工作）、Devoteam（研究融合基础设施）和Netcenter Informática LTDA（研究网络设备）。

本书第1章由Luís Henrique M. K. Costa撰写，讨论虚拟化技术，基本上而言，这使我们可共享计算资源，即将一个物理计算环境切片为相互隔离的虚拟计算环境。本章描述了Xen、VMware和OpenVZ虚拟化的主要特征并识别出它们的性能折中。就一台虚拟路由器所用的资源——CPU、RAM内存、硬盘和网络而言，给出虚拟化工具的性能结果。感谢Marcelo Duffles Donato Moreira、Carlo Fragni、Diogo Menezes Ferrazani Mattos和Lyno Henrique Gonçalves Ferraz，是他们定义了基准，并

实施了性能测试。

第2章由Miguel Elias M. Campista撰写，详细描述了Xen和OpenFlow虚拟化平台，并给出了二者的性能分析。选择这两个平台作为Horizon项目中所开发新提案的基础。本章也定义了网络虚拟化基础设施必须提供的原语，这就使引导平面可管理虚拟网络单元。感谢Natalia Castro Fernandes、Marcelo Duffles Donato Moreira、Lyno Henrique Gonçalves Ferraz、Rodrigo de Souza Couto、Hugo Eiji Tibana Carvalho，是他们定义了接口，并实施了试验。

第3章由Igor M. Moraes撰写，给出了在前一章讨论的两种平台的管理工具。为了控制和管理网络单元，定义了网络虚拟化基础设施必须提供的5个原语：instantiate（实例化）、delete（删除）、migrate（迁移）、monitor（监测）和set（设置）。为了验证概念，使用针对两种平台提出的接口，设计和开发了Xen平台的一个原型和OpenFlow平台的另一个原型。感谢Diogo Menezes Ferrazani Mattos、Lyno Henrique Gonçalves Ferraz、Pedro Silveira Pisa、Hugo Eiji Tibana Carvalho、Natalia Castro Fernandes、Daniel José da Silva Neto、Leonardo Pais Cardoso、Victor Pereira da Costa、Victor Torres da Costa、Rodrigo de Souza和Rafael dos Santos Alves，他们是工具的主要开发人员，并实施了试验。

第4章由Edmundo R. M. Madeira和Guy Pujolle撰写，描述了语境感知的技术和多智能体系统。引导系统是基于多智能体范型、以一种分布式方式开发的，目的是增加网络的规模扩展性。由此，给出了构造智能体的三个平台。

第5章由Miguel Elias M. Campista撰写，讨论虚拟化联网的现有控制算法。本章也分析了使用Xen作为虚拟化工具进行报文转发的主要挑战，并详细描述了虚拟网络局部控制的一种提案。在每个物理节点内，这个提案给出了虚拟网络隔离，确保了每个虚拟网络获得的服务水平，即使存在行为不当的虚拟网络的情况下也是如此。在本章描述的称为XNetMon的安全虚拟网络监测器，是由Natalia Castro Fernandes和Otto Carlos Muniz Bandeira Duarte提出和评估的。

第6章由Edmundo R. M. Madeira和Nelson Luís S. da Fonseca撰写，描述了引导系统。思路是引入一种自治系统，以此处理通信网络日渐增长的复杂性，从处理需要人类干预的任务中解放所需的网络管理员，这些任务如设置管理策略和提升任务的自动化程度——系统配置和优化、灾难恢复和安全。给出了一个多智能体自管理原型。试验是由Carlos Roberto Senna和Daniel Macêdo Batista实施的。

第7章由Otto Carlos M. B. Duarte撰写，讨论管理和控制功能。在监测和得到使用概要之后，知识平面使用预测机制，预测式地检测在虚拟网络配置中更新的必要性。知识平面存储信息，协助管理决策并执行网络维护。模糊控制方案是由Hugo Eiji Tibana Carvalho提出并评估的，ADAGA方案是由Pedro Silveira Pisa提出并评估的。

第8章由Otto Carlos M. B. Duarte撰写，详细地描述了用于系统架构的虚拟化

技术。基于 Xen 的路由器、OpenFlow 交换机和称为 XenFlow 的这二者的组合体，被用于集成机器和网络虚拟化技术。XenFlow 的关键思路是使用 OpenFlow 管理流，同时也用于支持没有报文丢失条件下的流迁移，并使用 Xen 提供路由和报文转发。XenFlow 是由 Diogo Menezes Ferrazani Mattos 和 Otto Carlos Muniz Bandeira Duarte 提出和评估的。

真诚感谢 Carlos José Pereira de Lucena、Firmo Freire、Djalmal Zeghlache、Jean-François Perrot、Thi-Mai-Trang Nguyen 和 Zahia Guessoum 等教授。也感谢 Marcelo Macedo Achá 和 Cláudio Marcelo Torres de Medeiros。真诚感谢原始思想和文章的葡萄牙的作者们，在本书中没有引用他们的文献，但确实是引入了他们的概念在本书中进行了讨论。最后，也感谢在 Horizon 项目中工作过并给出许多建设性的和深邃评论的所有人：Alessandra Yoko Portella、André Costa Drummond、Andrés Felipe Murillo Piedrahita、Callebe Trindade Gomes、Camila Patrícia Bazílio Nunes、Carlo Fragni、Carlos Roberto Senna、Claudia Susie C. Rodrigues、Cláudio Siqueira Carvalho、Daniel José da Silva Neto、Daniel Macêdo Batista、Diogo Menezes Ferrazani Mattos、Eduardo Rizzo Soares Mendes de Albuquerque、Elder José Reioli Cirilo、Elysio Mendes Nogueira、Esteban Rodriguez Brljevich、Fabian Nicolaas Christiaan van't Hooft、Filipe Pacheco Bueno Muniz Barretto、Gustavo Bittencourt Figueiredo、Gustavo Prado Alkmim、Hugo Eiji Tibana Carvalho、Igor Drummond Alvarenga、Ilhem Fejjari、Ingrid Oliveira de Nunes、Jessica dos Santos Vieira、João Carlos Espiúca Monteiro、João Vitor Torres、Juliana de Santi、Laura Gomes Panzariello、Leonardo Gardel Valverde、Leonardo Pais Cardoso、Luciano Vargas dos Santos、Lucas Henrique Mauricio、Lyno Henrique Gonçalves Ferraz、Marcelo Duffles Donato Moreira、Martin Andreoni Lopez、Milton Aparecido Soares Filho、Natalia Castro Fernandes、Neumar Costa Malheiros、Nilson Carvalho Silva Junior、Othmen Braham、Pedro Cariello Botelho、Pedro Henrique Valverde Guimarães、Pedro Silveira Pisa、Rafael de Oliveira Faria、Rafael dos Santos Alves、Raphael Rocha dos Santos、Renan Araujo Lage、Renato Teixeira Resende da Silva、Ricardo Batista Freitas、Rodrigo de Souza Couto、Sávio Rodrigues Antunes dos Santos Rosa、Sylvain Ductor、Tiago Noronha Ferreira、Tiago Salviano Calmon、Thiago Valentin de Oliveira、Victor Pereira da Costa 和 Victor Torres da Costa。

Otto Carlos M. B Duarte

Guy Pujolle

目　录

第1章 虚 拟 化

在本书中，将焦点放在基于多元化方法的一种新颖互联网架构上。一个多元化架构的例子如图 1.1 所示。在图 1.1 中，每个路由器层代表具有独立协议栈的一个不同网络，它们共享来自底层处基础网络设施的各种资源。虚拟化是使这样一种多元化架构成为可能的一项关键技术。虚拟化是这样一项技术，基本上说，它支持计算资源的共享［POP 74］。虚拟化将一个真实计算环境分成虚拟计算环境，这些环境是相互隔离的，并按照从真实的非虚拟化的环境中所期望的那样，与上面的计算层交互。一个虚拟化环境和一个非虚拟化环境之间的比较如图 1.2 所示。图的左手侧给出一个传统的计算环境，其中各项应用是在一个操作系统（OS）之上执行的，操作系统控制基础硬件。在图的右手侧，给出一个虚拟化环境，其中一个虚拟化层支持多个 OS 并行运行，每个 OS 都有其自己的应用，并控制它们到硬件的访问。当处理虚拟网络时，考虑路由器资源，如处理器、内存、硬盘、队列和带宽，这和计算环境虚拟化时是一样的。一个虚拟路由器和链路集合被称作一个虚拟网络。因此，使用虚拟化技术，可有多个并行的虚拟网络，每个网络都有一个特定的网络协议栈，共享单个物理网络基础设施，如图 1.1 所示。

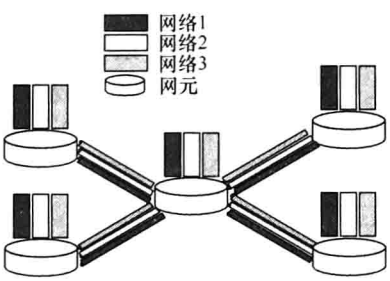


图 1.1 多元化架构范例

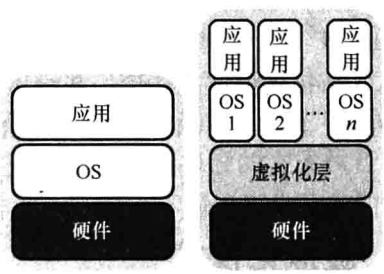


图 1.2 虚拟化环境范例

虚拟化普遍实现为称为 hypervisor 的一个软件层，它负责在多个虚拟环境或虚拟机（Virtual Machine，VM）之间复用计算资源。每个 VM 运行在 hypervisor 之上，由 hypervisor 控制到物理资源的访问。存在不同的 hypervisor 和虚拟化技术。本章给出最普遍的虚拟化工具（Xen［BAR 03，CHI 08］、VMware［VMW 07a］和 OpenVZ［KOL 06］）的主要特征概述并识别性能折中问题。就一台虚拟路由器的所关注资源方面［中央处理单元（CPU）、随机读取内存（RAM）内存、硬盘和网络］，该项研究比较了虚拟化工具的性能。虚拟路由器使用 CPU 处理到达报文，并依据转发表路由报文。使用 RAM 存储转发表。硬盘的主要用途是存储 VM 映像。使用

网络资源转发报文，这是一台路由器的主要任务。对于虚拟路由器的正常操作，CPU、RAM 和网络是虚拟化额外负担的最敏感资源。磁盘性能额外负担是人们所关注的，原因是它影响新路由器的实例化和虚拟路由器的迁移。为更好地理解由这种工具引入的额外负担，只要可能的情况，也给出原生的性能。

因为虚拟化可导致时间敏感应用的误操作（malfunction）［VMW 08］，所以使用与相关工作中所用技术［XEN 07，XMW 07b］不同的一种技术。将结果基于这样的时间，即一个虚拟化系统完成一项任务所需的时间，是从一个外部非虚拟化计算机测量得到的。

实施了两种类型的试验。第一种类型的试验目标是分析由额外层诱发的性能损失（hypervisor）是由虚拟化工具引入的。为取得这个目标，在第一种类型的试验中，仅有一个 VM 运行在虚拟化软件之上。将原生 Linux 的性能与 Xen、VMware 和 OpenVZ 虚拟化软件的性能进行了比较。结果表明，Xen 是基于 PC 的路由器虚拟化的一种良好适用物（fit），具有可接受的虚拟化额外负担（这在 1.4.6 节做了展示），支持 hypervisor 修改，是开源的，并提供虚拟路由器灵活性，原因是它具有一个虚拟硬件接口，支持在不同虚拟路由器中使用不同的 OS。另外，单个 VM 试验为第二种类型的试验（处理多个 VM）提供一个基线。第二个试验集合深入探讨所选中的虚拟化工具如何随着并行运行的 VM 数量而规模扩展的。这些种类的测试有如下目标，即澄清实例化的多个 VM（消耗同样的资源）如何影响总体性能，且虚拟化工具如何处理公平性。在这种类型的测试中，也验证各 VM 间 CPU 核分配的不同方案如何影响总体性能。

在 1.1 节，给出测试虚拟化工具所用的技术。在 1.2 节，详细描述虚拟化工具。1.3 节描述测试方法论，并给出测试床。1.4 节给出虚拟化工具比较中使用的基准测试，1.4.6 节和 1.4.7 节给出性能比较结果。最后，在 1.5 节给出本章的结语。

1.1 虚拟化技术

为了更好地理解所比较的虚拟化工具，重要的是识别不同的虚拟化技术。本节描述 Xen、VMware 和 OpenVZ 使用的概念和技术。

虚拟化有许多不同的定义，但就虚拟化是共享计算资源、在虚拟环境之间赋予一定程度的隔离的一项技术方面，这些定义是一致的。依据 Popek 和 Goldberg［POP 74］，经典定义是，虚拟化是提供 VM 的一项技术，这些 VM 是底层硬件的高效被隔离复制。如今，这个概念不仅可在硬件上一般化，而且可在任何计算资源上一般化，作为一个 OS 内核或编程语言（如 Java 和 C#）使用的 VM 抽象。

从虚拟目标中出现了几项挑战。第一个挑战是调用所有虚拟环境访问相同的底层计算资源。对于硬件虚拟化，共享资源是 CPU、RAM、存储和网络。RAM 共享

访问可以不同方式完成。虚拟环境可被赋予到虚拟内存空间（可被转换为物理RAM，这和OS的做法相同）的访问权限。另一种方法是让VM知道它们的虚拟化特征，并允许它们在由hypervisor指派一个区域内存片之后，直接访问内存。CPU共享可以几种方式完成，并可使用诸如轮转法、加权的轮转法、应需分配和其他方法等机制做到。一般而言，输入/输出（I/O）可使用存储交换数据的缓冲这样一种统一的方式加以处理，这些数据是在物理外设和虚拟外设之间复用和解复用的。

试验使用商用的基于x86的硬件，这对虚拟化施加额外挑战。20世纪70年代，在虚拟化开发的开始阶段，所设计硬件都是支持虚拟化的。大型主机有这样的指令集，其中处理资源分配和使用的所有指令都是特权指令，即程序要求一定的CPU特权执行等级。在这些CPU架构中，可使用称作“去特权”［ADA 06］法的一项技术做到硬件虚拟化，其中虚拟化OS是在一个非特权语境中执行的，目的是无论何时资源分配或使用指令将被执行时，就产生陷阱。出于这个原因，hypervisor将截获陷阱，并以一种对其他VM安全的方式来模拟所需资源的分配或使用。依据Popek和Goldberg［POP 74］的说法，为硬件虚拟化构造一个hypervisor存在三个需求：①效率，这意味着来自虚拟CPU（vCPU）之指令集的一个大型子组应该直接在真实CPU中执行，不需要来自hypervisor的任何干预；②资源控制，这意味着hypervisor必须对所有资源具有完全的控制；③等价性，这意味着hypervisor必须向虚拟环境提供等价于原始接口的一个虚拟接口。20世纪70年代的大型主机将有利于构造hypervisor，原因是使用去特权技术就可做到VM隔离。对于基于x86的硬件，情况就不是这样的，原因是出于优化目的，基于x86的硬件指令集具有访问共享资源的指令，但不要求一个特权化的语境。此外，基于x86的指令集包含这样的一组指令，它们被归类为对特权等级是敏感的，其中指令以取决于当前特权等级的一种不同方式执行。如果一个去特权的OS执行一条敏感的指令，它将悄无声息地失败，原因是它不会产生一条陷阱，而且它也不以OS所拟设的方式执行。为处理这些基于x86的硬件问题，存在几项解决方法，在描述不同虚拟化技术的后面各节中将给出。

1.1.1 完全虚拟化

完全虚拟化是这样一种虚拟化技术，其中虚拟化所有的原始接口，且输出给虚拟环境的各接口与原始接口完全相同。在这种方法中，guest OS（寄居OS），即驻留在VM内的OS，不需要修改，直接在VM内执行。为处理来自基于x86硬件平台的敏感指令，可使用不同技术。一种著名的技术是二进制翻译。二进制翻译检查要执行的代码，搜索存在问题的指令，并将它们替换为模拟期望性能的那些指令。二进制翻译的优势是，它支持应用和OS在不做修改的条件下加以使用。不过，二进制翻译诱发高的CPU额外负担，因为所有执行的代码都必须做检查，且存在问题的指令必须在运行时加以替换。

最近，存在来自主要硬件制造商的大力投入来优化虚拟化。服务器合并法使用虚拟化将具有空闲容量的几台服务器替换到单个硬件，该硬件具有较高的利用率，其中执行几个虚拟服务器。服务器合并法已经成为共性实践，它在主要公司中削减设备和维护成本。出于这个原因，AMD 和 Intel 都为在现代 CPU 中提供更高效的虚拟化支持开发了各项技术。Intel 虚拟化技术（IVT）和 AMD 虚拟化（AMD-V）都为完全虚拟化提供较佳的性能，方法是引入两种新的操作模式：root 和 non-root。root 操作模式由 hypervisor 使用，类似于常规 CPU 操作，提供完全的 CPU 控制和传统的特权等级的四个环。non-root 模式是用于 VM 的执行的。在这种模式中，CPU 也提供特权等级的四个环，寄居 OS 不再在一个去特权化环中执行，而是在环 0 中执行，这个环就是为之设计的。无论何时寄居 OS 执行一条存在问题的指令，CPU 就产生一个陷阱，并将控制返给 hypervisor，来处理这条指令。采用这种 CPU 支持，二进制翻译就不再是必要的，且完全虚拟化 hypervisor 就极大地增加了它们的性能。

1.1.2 半虚拟化

半虚拟化是这样一种虚拟化技术，其中寄居 OS 与 hypervisor 协作得到较佳的性能。在半虚拟化中，修改寄居 OS，无论何时执行一条存在问题的指令，就调用 hypervisor。敏感指令被替换为调用 hypervisor 的一条虚拟化感知指令。出于这个原因，hypervisor 不需要为存在问题的指令而监测 VM 执行，相比于使用二进制翻译的完全虚拟化，这极大地降低了额外负担。折中代价是，OS 必须被修改和重新编译，产生半虚拟化的 OS 映像。为进行 hypervisor 调用，需要半虚拟化的 OS 映像。这阻碍了遗留 OS 的虚拟化，并要求 OS 开发商的协作才能完成。

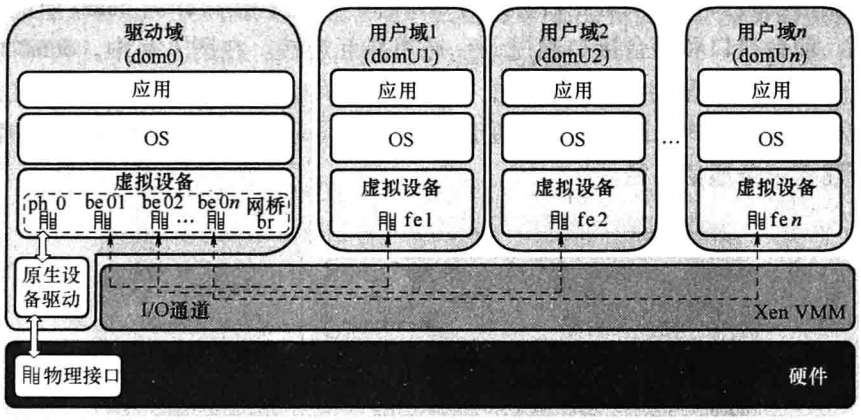
1.2 虚拟化工具

本节描述主要的虚拟化技术，并给出 Xen、VMware 和 OpenVZ 的性能评估结果。

1.2.1 Xen

Xen 是一个开源 hypervisor，提出它，为的是运行在商用硬件平台上，它使用半虚拟化技术［BAR 03］。Xen 使我们可在单个物理机器上同时运行多个 VM。Xen 架构由一个 hypervisor（位于物理硬件之上）和 hypervisor 之上的几个 VM 组成，如图 1.3 所示。每个 VM 可有其自己的 OS 和应用。hypervisor 控制到硬件的访问，也管理由各 VM 共享的可用资源。另外，为提供可靠的和高效的硬件支持［EGI 07］，设备驱动被放在一个孤立的 VM［称作 Domain 0（dom0）］中。因为 dom0 具有对物理机器硬件的完整访问权限，所以相比其他 VM，它有特权，参见用户域（domUs）。另外，domUs 有虚拟驱动，称作前台驱动，它与位于 dom0 中的后台驱

动通信来访问物理硬件。接下来简短地解释 Xen 如何将所关注的每种机器资源（处理器、内存和 I/O 设备）虚拟化为一台虚拟路由器。



fe-前台接口，be-后台接口，ph-物理接口

图 1.3 Xen 架构

通过将 vCPU 指派给 VM，Xen 对处理器实施虚拟化。vCPU 是在每个 VM 内运行进程可看到的各 CPU。Xen hypervisor 实现一个 CPU 调度器，动态地将一个物理 CPU 在某个时段期间映射到每个 vCPU。Xen 所使用版本（3.2）的默认调度器是信用调度器，实施一个比例性的 CPU 共享。依据指派给各 VM 的权重，信用调度器将 CPU 资源分配给每个 VM（或更具体而言，是分配给每个 vCPU）。信用调度器在对称多处理（SMP）主机上也可以是工作守恒的。这意味着调度器允许物理 CPU 运行在 100% 处，如果任何 VM 有工作要做。在一个工作守恒的调度器中，对一个 VM 可使用的 CPU 资源量是没有限制的。

在 Xen 中的内存分配目前是以静态方式完成的。每个 VM 接收定量的内存空间，这是在其创建时刻被指定的。另外，为要求来自 hypervisor 的最小干预，各 VM 负责分配和管理硬件页表的相应部分。当一个 VM 每次要求一个新的页表时，它就从其自己的内存空间中分配并初始化一个页，且将之注册到 Xen hypervisor，后者负责确保隔离性。

在 Xen 中，使用共享内存异步缓冲描述符环，来自 I/O 设备的数据被传递进出每个 VM。Xen hypervisor 的任务是实施确认检查。例如，检查各缓冲被包含在一个 VM 内存空间内。通过使用它的原生设备驱动，dom0 直接访问 I/O 设备，同时代表 domUs 实施 I/O 操作。另外，domUs 使用其后台驱动从 dom0 请求设备访问[MEN 06]。一种特殊情形的 I/O 虚拟化是网络虚拟化，它负责将来自物理接口的到达报文解复用到各 VM，同时复用由各 VM 产生的外发报文。图 1.4 所示为 Xen 使用的默认网络架构。对于每个 domU，Xen 创建由这个 domU 要求的虚拟网络接口。这些接口也称作前台接口，domUs 用之进行其所有的网络通信。此外，在