计算机研究与发展
Journal of Computer Research and Development
DOI:10.7544/issn1000-1239.2015.20140207
52(9):2069-2082,2015

未来互联网虚拟化研究

余 涛 毕 军 吴建平
(清华大学信息网络科学与网络空间研究院 北京 100084)
(清华信息科学与技术国家实验室(清华大学) 北京 100084)
(junbi@tsinghua.edu.cn)

Research on the Virtualization of Future Internet

Yu Tao, Bi Jun, and Wu Jianping
(*Institute for the Network Sciences and Cyberspace, Tsinghua University, Beijing* 100084)
(*Tsinghua National Laboratory for Information Science and Technology (Tsinghua University), Beijing* 100084)

Abstract Current Internet is designed based on end to end principle and co-established by lots of Internet service providers with different objects and policies. In order to update Internet architecture, it needs to reach a consensus among almost all of them, which makes the direct deployment of radically new architecture and protocols on Internet nearly impossible. To fend off the ossification of Internet architecture, network virtualization is proposed to add diversity in the future Internet. By introducing various architectures on a common substrate network, Internet virtualization can promote Internet innovation and encourage the emergence of huge kinds of new applications. In this paper we firstly describe the application context of network virtualization and the general virtualization method of traditional networks, then setup a classification framework for various related researches on Internet virtualization. After that we explore them one by one from the perspectives of both Internet architecture and test bed, and summarize the development trend of research on future Internet virtualization. We conclude that network virtualization is an indispensable part of future Internet and takes two roles in it. One is for underlay network, the other is for each future Internet paradigm, and both of them are required for future Internet to realize its design goals.

Key words network virtualization; virtual network; network virtualization architecture; software defined network (SDN); future Internet architecture

摘 要 目前的互联网基于“端到端”原则设计,并且由许多目标与策略迥异的互联网服务提供商共同创建和维护.若想升级互联网体系结构,需要得到他们全体的一致同意,因此几乎不可能在互联网上部署全新的网络结构和协议.为了避免互联网结构出现僵化,人们提出可用网络虚拟化方法增加互联网范式的多样性.通过在一个公共的基底网络上引入多种不同的互联网结构,互联网虚拟化可以促进互联网创新并推动更多种类的新应用出现.分别从互联网体系结构和实验平台2个方面,对过去和当前正在进行的有关未来互联网的虚拟化研究进行分析、归类和概述,并总结未来互联网虚拟化研究的发展趋势.

关键词 网络虚拟化;虚拟网络;网络虚拟化结构;软件定义网络;未来互联网体系结构

中图法分类号 TP393

收稿日期:2014-03-11;**修回日期**:2014-06-04

基金项目:国家“九七三”重点基础研究发展计划基金项目(2009CB320501);国家“八六三”高技术研究发展计划基金项目(2013AA013505);国家自然科学基金项目(61073172,61472213)

互联网在其诞生后的40年里取得了显著的成就,它不但支持丰富的分布式应用,而且支持大量不同种类的网络技术.随着互联网应用的推广,人们对它的要求也远远超出其最初的设计目的.尽管针对某些新要求已经有各种目的单一并可逐步更新的改良方法,然而,暂且不论这些逐步改进能否得到实际实施并在互联网上真正部署,仅从方法论上看,这样做长久下去会对互联网的结构清晰性、可靠性、可管理性和可诊断性造成危害.因此在研究界有一种看法认为有必要从头开始重新设计互联网,并提出可使用网络虚拟化方法增加互联网范式的多样性[1-2].

正如其他领域的虚拟化技术那样,网络虚拟化技术将组成物理网络的硬件与其上运行的逻辑网络解耦合,可以在同样的网络硬件上创建特征差异巨大的不同逻辑网络,并允许它们同时存在.此外,虚拟化物理网络资源所引入的抽象使网络管理者能够以异常灵活和机动的方式管理和修改网络.它不但可以为研究人员提供功能强大、可灵活定制的实验环境,而且可提供一种部署环境应用于ISP、机关、企业甚至家庭中.其中对互联网的虚拟化可以使ISP动态地生成多个不同结构的虚拟互联网,它们可以同时存在并且相互隔离.通过有效分享来自多个基础设施提供者的底层网络资源,ISP可以为用户部署和管理定制的、端到端的互联网服务.这种动态的互联网环境不但可以增强ISP部署不同互联网结构的能力,而且可以不受当前互联网结构的限制、为不同的终端用户提供多样化的服务,从而消除在当前互联网中存在的部署僵局和网络服务僵化现象[3],使未来的互联网能够以演进的方式长期持续发展.

目前网络虚拟化研究已引起人们的广泛兴趣和重视,它不但在工业界得到大力开展[4],而且在学术界也成为一个单独的研究课题(research topic)[5].此外一些标准化组织也开始了对网络虚拟化的规范工作,例如ITU-T在2009年设立了未来网络焦点组(Focus Group),并确立网络虚拟化为该工作组的一个基础研究课题;IRTF也在2010年成立了虚拟网络研究组VNRG,专门研究网络虚拟化.然而随着网络虚拟化研究的广泛开展,"网络虚拟化"这个术语本身被人为地赋予多重含义(overloaded connotation),从而给人们理解网络虚拟化造成困难.例如思科公司在一份技术文档中将网络虚拟化定义为在一个公共的企业物理网络设施上创建相互隔离的逻辑网络[6];文献[7]将网络虚拟化定义为一种技术,它的目的是通过虚拟化将计算和网络资源进行分割并分配给虚拟网络,从而使它们容纳多个独立并且可编程的虚拟网络;文献[8]则用虚拟化之后的网络状态描述网络虚拟化,认为网络虚拟化是一种环境,它允许多个服务提供者通过分享基础设施提供者的资源在基底网络上同时创建多个异构的虚拟网络,并保持彼此隔离;文献[4]将网络虚拟化描述为对一组网络资源进行任意形式的分割或组合,并且在将这组经过抽象的资源提供给多个用户时,每个用户通过自己得到的那部分经过分割或组合后的资源,具有与其他用户分开的唯一网络视图.这组网络资源既可以是基本的(例如节点和链路),也可以是派生的(例如网络拓扑),并且可以被递归地虚拟化.

本文综述未来互联网虚拟化研究,与以前的网络虚拟化研究综述不同,它专注于互联网的虚拟化,是对以前综述的小范围深入研究和补充.本文目的并非试图对"互联网虚拟化"概念进行标准化的定义,而是从驱动网络虚拟化的场景出发,通过分析和总结有关未来互联网虚拟化的代表性研究项目,期望读者能够激发出新的互联网设计和虚拟化方法,从而使未来互联网满足已知的应用需要并适应未来的发展要求.

1 背 景

本节举例说明网络虚拟化的需求场景,并对传统网络虚拟化的一般性方法进行介绍.

1.1 网络虚拟化驱动场景

1) 推动网络虚拟化的原因之一是有效分摊物理网络的成本.由于部署物理网络基础设施花费巨大,许多服务提供者因为资源有限而被阻止在市场之外.通过网络虚拟化技术对现有的网络基础设施进行资源分割并将切分后的各个资源片组合成虚拟网络租借给服务提供者运营,将大大降低服务提供者的市场进入成本.

2) 为了在现有的网络基础设施上提供增值服务,可通过网络虚拟化方法在虚拟网络中引入新技术,或对已有的协议进行定制使其服务于某个特殊目的.此外,虚拟网络还可以通过递归与继承提供细分服务.

3) 为了使多个异构的网络基础设施能够相互合作,即进行水平互联,可通过抽象层(虚拟化层)将它们统一为一个虚拟网络.

图1分别描绘了上述3种场景:

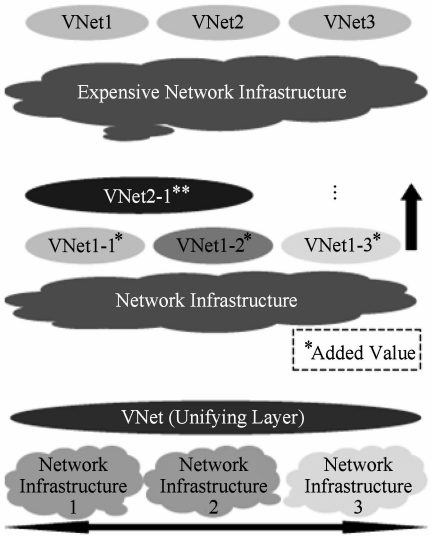

Fig. 1　Network virtualization driven context.

图 1　网络虚拟化驱动场景

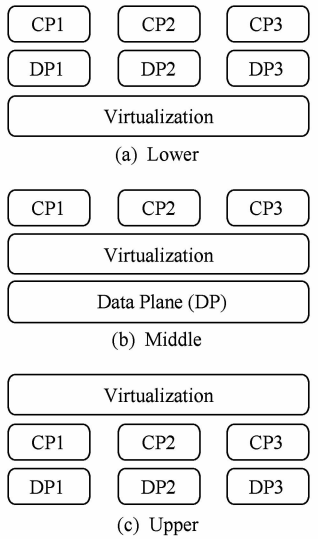

Fig. 2　Network virtualization method for traditional networks.

图 2　传统设计网络的虚拟化方法

1.2　传统网络虚拟化的基本方法

虚拟化是对资源的一种抽象,这种抽象通常由一个软件层实现,用于提供与真实资源接口相似的虚拟资源接口.在网络中有 2 种资源:1)网元;2)链路.在传统的分布式网络设计中,网元结构同时包含控制平面和数据平面,由于链路通常只是在网元之间提供物理连接,因此根据抽象层在网元中放置的位置,主要有 3 种网络虚拟化方法:1)虚拟化层位于控制平面和数据平面之下,在这种情况下网元的数据平面和控制平面都被虚拟化,每个虚拟网元可以实现自己的数据平面和控制平面,网络的可编程度最高;2)虚拟化层位于控制平面和数据平面之间,在这种情况下只有网元的控制平面被虚拟化,每个虚拟网络运行各自的控制软件但共同分享网元的数据平面;3)虚拟化层位于控制平面和数据平面之上,这种情况可认为是将多个网元以及它们之间的链路进行组合抽象,即将由它们共同组成的网络抽象成一个网元,或将由它们组成的多个异构网络抽象成一个网络.虚拟化层的位置示意如图 2 所示.

值得提出的是,上述第 2 种方法,即只虚拟化网元控制平面的方法还可根据虚拟网元在数据平面的隔离等级进一步细分为多个子类.如果同一网元上的各个虚拟网元在该网元的数据平面存在强隔离,那么该网元上的每个虚拟控制平面只能访问其数据平面的一部分,并且不会干扰该网元上其他虚拟控制平面对数据平面的访问.另一方面,如果网元的数据平面对该网元上的所有虚拟控制平面不作区分和限制而任由它们分享,那么在虚拟控制平面之间会存在相互影响,这种情况被称为弱隔离.

2　未来互联网虚拟化设计目标

根据人们对互联网虚拟化的期望,互联网虚拟化设计的好坏将直接影响到未来互联网能否全面解决当前互联网面临的问题,并足以应对目前还无法预测的未来人们对互联网的要求.因此,互联网虚拟化设计必须满足一系列目标,它们不但可以用来指导未来互联网结构和协议的设计,而且可以用来评价不同的互联网虚拟化方案.

1) 灵活性.未来互联网中的虚拟网络必须具有全面的灵活性,每个服务提供者可以任意设计虚拟网络的拓扑、路由、转发以及控制协议,而不受底层物理网络和其他虚拟网络的制约.

2) 可管理性.由于在未来互联网中服务提供者与基础设施提供者彼此独立,因此互联网管理任务需要根据参与角色模块化,并且有必要在不同的参与者之间引入可问责性.在未来互联网中,基础设施提供者负责物理网元的运营管理,并提供资源访问接口,服务提供者则对虚拟网络具备端到端的控制能力,负责虚拟网络的运营并为终端用户提供服务,而且不同服务提供者的网管视图可以各不相同并且不会相互影响.实现对这种网络结构的管理需要合理划分和设计网管任务,有效地分散管理并做到责任分明.

3) 可扩展性.多个虚拟网络并存是未来互联网的基本特征之一,在不影响已存在虚拟网络性能的情况下,基础设施提供者对物理资源的虚拟化应该能够可扩展,以便支持尽可能多的虚拟网络.

4) 网络隔离.由于网络协议经常存在实现缺陷和配置错误,以及为了防止从某个虚拟网络内部发动针对其他虚拟网络的攻击,互联网虚拟化技术必须确保虚拟网络彼此隔离,将每个虚拟网络可能出现的风险限制在该虚拟网络内部,从而提高未来互联网整体的容错性、安全性和私密性.

5) 可编程性.为了保证虚拟网络的灵活性与可管理性,未来互联网的网元必须支持可编程,但对网元应具备何种程度的可编程性以及如何在网元中实现它目前仍有争议.如何在未来互联网中设计一种方便、有效而且安全的可编程方案,实现基础设施提供者与服务提供者的双赢,是实现未来互联网虚拟化的关键目标之一.

6) 网络实验与部署方便.互联网虚拟化研究的动机之一就是直接在未来互联网上对新型的网络结构进行实验和部署.实际上正是由于搭建 Overlay 网络和网络实验平台的方法无法提供一个真实、可靠和大规模的实验环境,才导致网络虚拟化研究的诞生.另外商业机构也希望能够借助网络虚拟化在现有网络上快速可靠地部署创新的服务以提升综合竞争力.因此未来互联网的虚拟化既要支持对新型互联网体系结构的实验验证,又能作为一种部署途径,快速方便地部署新型结构的互联网.

7) 异构性.未来互联网的异构性具有 2 个方面的内容:①底层物理网络的异构性;②端到端虚拟网络的异构性.其虚拟化结构必须允许服务提供者可以不受限制地在任何异构的物理网络上组建任何异构的虚拟网络,同时底层的物理网络必须能够支持任何异构的互联网协议和算法.

8) 对遗留技术的支持.任何新技术在部署之前都要考虑对遗留技术的支持或后向兼容性,在概念上互联网虚拟化可以在未来互联网中集成当代互联网;但未来互联网的虚拟化结构能否有效做到这一点以及如何做到,仍是一个重要的研究挑战.

3 未来互联网虚拟化研究分类框架

传统的互联网也叫“网间网”,它覆盖(overlay)在各种类型的网络之上并实现它们之间的相互联通,位于 OSI 参考模型结构的第 3 层及之上.在未来互联网研究中,被互联的各种网络称为基底网络(也有人称之为基础设施网络,本文对这 2 个概念不加区分),它们可以是 2 层网络也可以是 3 层网络.这些基底网络还可以按目前的社会结构划分,将它们分成用户网络和服务提供者网络(即运营商网络).因此,根据网络虚拟化发生的范围是位于互联网结构内还是基底网络结构内,可将未来互联网的虚拟化研究分为 2 类:1)互联网体系结构的虚拟化研究;2)网络实验平台(也即未来的基底网络或网络基础设施)的虚拟化研究.

在对互联网体系结构的虚拟化研究中,关于网络虚拟化在未来互联网体系结构中的作用,目前还存在 2 种不同的观点:结构单元论[9]和结构多元论[10].“结构单元论者”认为网络结构是通用的(general-purpose),能够为现存的所有应用提供一个合适的平台,并能设法支持未来应用的要求.当网络结构不足以支持某个给定的应用时,网络开发者应当为它设计一个合适的、具有次优属性的 Overlay,或者等待互联网采用一种新的更加优越的网络结构;“结构多元论者”却认为未来互联网将允许多种网络结构同时存在,新的互联网结构可以在现有的网络基底上随时出现或消失,并把这些具有不同结构的互联网的动态集合称作未来互联网.由于网络虚拟化支撑了新互联网结构的出现,而新互联网结构可以提供新服务以满足人们不断增加的需求,因此他们认为网络虚拟化是未来互联网结构的一种基本属性,并相信它不但能够彻底根除互联网结构的僵化,而且可以促进网络升级和应用创新.

根据结构单元论的观点,可依照网络结构能否与上一代互联网结构兼容,分为可演进结构的虚拟化研究和全新(clean slate)结构的虚拟化研究.在单元可演进结构的互联网虚拟化研究中,由于互联网到目前为止只发展了一代,也即对当前互联网结构的虚拟化研究中,可根据网络虚拟化发生的层次进行区分.对于单元 clean slate 结构的虚拟化研究,由于提出完整体系结构并考虑到网络虚拟化的研究较少以及这类结构的创新性,对它们不再细分.

在结构多元论的观点中,未来互联网虚拟化研究的首要任务是建立一个一般性的、坚实的互联网虚拟化结构(virtualization architecture),各个互联网的结构、协议甚至网络虚拟化则属于各自网络体系结构的研究内容,并非其考虑的重点,而且支撑未来互联网结构的底层物理网络属于网络实验平台的

研究内容,它们也可通过这些项目去研究.据此可知,结构多元论中的互联网虚拟化研究涵盖了结构单元论中的互联网虚拟化研究,即属于未来互联网多种联网范式之一的各个互联网体系结构的研究仍可包含网络虚拟化研究.另外由于网络实验平台需要同时支持多个并行的网络实验,对网络实验平台的研究也包含虚拟化结构的研究,因此在下文的趋势总结中,我们把结构多元论中的网络虚拟化研究与网络实验平台的虚拟化研究放在一起讨论.

关于网络实验平台的虚拟化研究,根据研究的时间秩序,早期的实验网络采用的是传统的网元控制平面和转发平面一体的分布式网络设计;近年来采用的则是软件定义网络(software defined network,SDN)[11]范式,它将网元的控制平面从网元结构中剥离并对整个网络采用集中控制平面,具备可编程性的网元通过开放接口接受来自集中控制平面的控制,并且只在转发平面进行分组转发.对于早期的实验网络,可根据它们对网络资源虚拟化的粒度进行区分.对于采用 SDN 范式的网络虚拟化研究,则可根据是基于 SDN 目前规范的虚拟化研究还是旨在扩展其规范的虚拟化研究进行区分.各种互联网虚拟化研究的分类如图 3 所示:

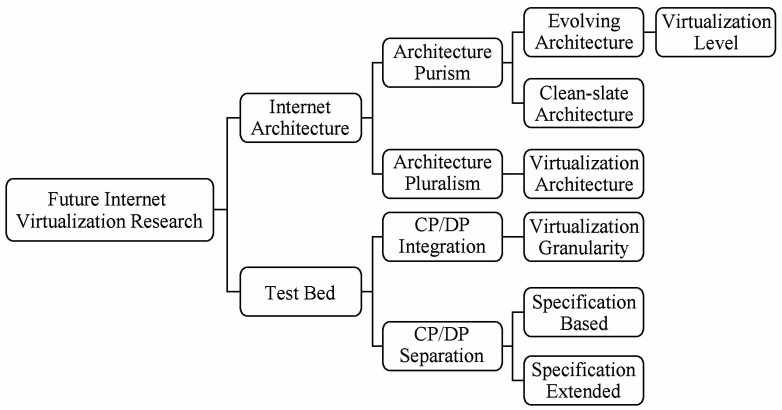

Fig. 3　Classification framework for researches on future Internet virtualization.

图 3　未来互联网虚拟化研究分类框架

4　互联网体系结构的虚拟化研究

本节我们根据研究者对未来互联网体系结构的看法,分别从单元可演进结构、单元 clean slate 结构和多元结构 3 个类别对已存在的互联网虚拟化研究项目进行介绍.

4.1　单元可演进结构的互联网虚拟化研究

1) 应用层——VIOLIN

VIOLIN(virtual Internetworking on overlay infrastructure)[12]是一种应用层上的虚拟网络结构,它在 Overlay 网络(例如 PlanetLab)上用软件生成各个虚拟网络,并使它们保持彼此隔离.VIOLIN 由虚拟路由器、虚拟局域网和虚拟主机 3 部分组成,它们都是 Overlay 网络主机上的软件.其中虚拟路由器和虚拟主机是运行在 Overlay 网络主机上的虚拟机,虚拟局域网则是由虚拟交换机连接多个虚拟主机形成,最后由虚拟路由器连接多个虚拟局域网形成整个虚拟网络.利用虚拟机技术的进步,VIOLIN 将虚拟网络的隔离等级从单个虚拟机节点扩展到了整个虚拟网络.VIOLIN 提供的网络隔离包括 4 类:管理隔离、地址空间和协议隔离、网络攻击和故障隔离、资源隔离.通过这些隔离手段,VIOLIN 能够提供一种受限、安全和专用的网络环境,可用于部署不可信的分布式应用或运行有风险的网络实验.

2) 网络层——X-Bone

X-Bone[13]是一种能够快速、自动地在 IP 网络上配置和管理 Overlay 网络的管理工具,并加强了 Overlay 网络的安全以及对它的监控.它由 Overlay 管理器、资源守护进程和组播控制协议 3 部分组成.Overlay 管理器控制 Overlay 网络的建立和配置,并提供一个可编程 API 供用户使用;资源守护进程运行在每个资源内部,执行访问控制、资源管理和安全防护等功能,用来协调对资源的使用;组播控制协议则用于找出可用资源的位置,提供管理控制和反馈以及宣告应用程序使用的通道.X-Bone 可视为在互联网上部署 Overlay 网络的工具,并能为多个 Overlay 网络提供隔离和资源分配机制.

这种网络层虚拟化方法后来又被进一步扩展并形成虚拟互联网(VI)结构[14].通过将主机、路由器、

链路和网络拓扑描述成网络资源,这种 VI 结构探索了虚拟化环境对网络资源的影响.

VI 是由虚拟主机、虚拟路由器和 IP 隧道链路构成的 IP 网络,它除了具有当前互联网的全部功能之外,还支持资源的动态发现、部署和监控等功能.另外,VI 还支持控制递归、网络递归和再访问,控制递归允许对网络进行分治管理,网络递归允许 VI 嵌套,再访问使单个互联网节点可以同时作为多个虚拟节点出现在同一个 VI 中.所有这些特性保证了 VI 的一致性,使 VI 的结构灵活通用并且功能强大.

4.2 单元 clean slate 结构的互联网虚拟化研究

1) RNA

通过研究网络分层以及协议栈与结构的关系,Touch 等人[15]提出一种递归的网络结构 RNA,它可以把虚拟化的网络环境自然地集成到这个新网络结构中.在 RNA 中,网络协议可以动态创建,由下而上实现一个新的网络环境.这样设计是因为网络协议栈运行在一个动态的环境中,因此它应该能够随时间发生相应的变化,例如为了给网络增添新的服务和功能需要有新的协议或重新实现原有协议;然而当前网络协议栈的设计却是静态的,并且由于最初的互联网设计方案认为网络服务应该在网络协议栈的低层实现,所以导致当前互联网的功能无法动态改变,新的协议与服务只能由用户在应用层上实现.

为了满足应用环境对网络协议的性能要求,RNA 开发了一种可以动态合成网络协议的网络结构.它的核心思想是使用单一、通用的元协议(meta protocol)[16]来实现一般的功能和服务,并提供手柄(knob)供这些服务根据元协议所在的层次进行调整.通过合成最优的网络协议栈实现网络的新性能,RNA 天然地支持动态定义的虚拟网络,而且由于动态合成过程需要跨层互动,RNA 还可避免在多个层次上重复实现服务.

2) Internet 3.0

Jain 等人[17-18]也提出一种 clean slate 结构 Internet 3.0,它允许虚拟网络依据策略建立在功能互相联合的多个提供者的资源之上,并可以进行优化以满足特定的应用要求.当前的互联网认为网络是一种"通信系统",即网络是一种分开的系统,它通过标准化接口为应用场景提供服务.而在 Internet 3.0 中,网络作为一种"通信范式"被认为是应用场景的一部分."虚拟网络"即定义在该观点之下,它是多元化和可编程的,而且允许动态配置从而有效地服务某个应用场景的特定要求.

4.3 多元结构的互联网虚拟化研究

1) CABO

CABO(concurrent architecture is better than one)[19]是一种允许多种网络体系结构并存的下一代互联网设计方案,它的核心思想是将网络基础设施与其提供的网络服务分开,从而打破互联网体系结构更新的僵局.基础设施提供者通过对基底设施虚拟化出租其设备,服务提供者利用这些设备部署自己的网络结构和协议并运营自己的网络服务以满足各种各样的应用要求,而且这些服务可以同时存在并保持彼此隔离.

CABO 还支持虚拟路由器从一个物理节点自动迁徙到另一个物理节点[20],增加网络可问责性以保证服务提供者的需求得到满足[21],并使用能迅速响应网络变化的可扩展的多层路由方案[22].在支持路由器可编程方面,CABO 与主动网络中的思想相似,但它不允许用户对网络编程,而是让服务提供者通过定制网络为最终用户提供端到端的服务.

2) 4WARD

4WARD 是欧盟第 7 次框架计划(FP7)下的未来互联网结构和设计项目(architecture and design for the future Internet)[23],它承诺通过电信级的网络资源虚拟化,实现公共平台上的多个网络并存,并提供虚拟网络的按需实例化,支持异构的虚拟网络在安全和可信商业环境中的可靠互操作.在 4WARD 的核心结构设计中,还考虑到了异构网络技术的虚拟化(例如有线和无线技术)、不同用户终端设备的虚拟化以及新型网络协议的虚拟化.

4WARD 在其商业模型中引入 3 种不同的角色:1)基础设施提供商,它管理底层的物理网络资源;2)虚拟网络提供商,它负责创建虚拟网络;3)虚拟网络运营商,它基于已创建的虚拟网络为用户提供服务.尽管 4WARD 与其他网络虚拟化项目存在很大程度上的相似性,但它有一个显著的特点,即承诺将网络虚拟化真正地带给最终用户,而不仅仅将它用于网络实验或实验平台.

5 网络实验平台的虚拟化研究

本节我们根据先后次序分别对网络实验平台的虚拟化研究进行介绍.

5.1 项目介绍

1) PlanetLab

PlanetLab[24]是基于 Overlay 网络的全球性实验平台,开发它的目的是为了设计、评估和部署分布

式的网络服务,并且能够同时服务于生产和科研.设计人员旨在创建一种面向服务的网络结构,能够把分布式系统与网络的优势结合起来.它的设计原则如下:1)支持资源切分(sliceability),即每个应用可获得 Overlay 网络的部分资源(slice)并在其上运行.slice 被定义为运行在 PlanetLab 节点上并通过互联网相互连接的虚拟机集合.节点资源切片(sliver)由运行在每个节点上的虚拟机监控器(VMM)分配和调度,用于创建一个分布式的虚拟环境.2)支持分散的控制结构,每个节点可以有自己的本地策略.3)管理服务分散.它没有集中的 Overlay 网络管理服务,而是将管理任务划分成运行在每个 slice 上的子服务并对其进行管理.4)Overlay 网络支持流行的编程接口(API),这能促进人们开发长期运行的真实服务,而不是只将它作为实验平台用于暂时的实验.

PlanetLab 上的每个节点有一个虚拟节点管理器(NM),它与一个中心控制模块(PLC)相连.PLC 作为可信的中间人,可根据节点主人指定的策略代表节点主人管理节点资源,为实验用户创建 slice 并管理 slice 的资源分配.PLC 支持直接和间接 2 种形式的 slice 实例化,通过它在每个节点上运行的叫做 pl_config 的 slice 创建服务.直接方式就是由 PLC 的前端直接与 pl_config 联系,创建相应的虚拟机并分配资源;在间接方式中,每个用户的 slice 创建代理向 PLC 请求一个“ticket”,这个 ticket 封装有在某个 PlanetLab 节点上创建虚拟机的特权以及分配给虚拟机的资源等信息.用户代理然后请求节点上的 pl_config 履行 ticket,为其创建一个 slice.slice 一旦产生,用户就可引导它(bootstrapping)开展自己的服务.

尽管 PlanetLab 很流行,但它的功能集中在节点的虚拟化和全局资源管理上,主要用于研究网络服务,用它研究网络结构和协议并不是很合适.

2) GENI

GENI(global environment for network innovation)[25]是美国国家自然科学基金会发起的研究计划,它基于从 PlanetLab 以及其他类似的实验平台中得到的经验,旨在创建一个开放、大规模和真实的全球性实验设施,用于测试和评估未来的互联网体系结构.它既可以在实验平台内传送其内部用户的真实流量,也可以与当前的互联网连接从而延伸到外部网络.与一般的实验平台不同的是,GENI 是一种通用的大规模实验实施,致力于为研究人员提供建立自定义网络的机会,并且对要验证的网络结构、服务和应用没有任何限制,能够在自然条件下使用真实的用户流量对任何网络结构进行实验.

GENI 的主要设计目标如下:支持资源分割实现资源分享,为研究人员提供的初始平台具有普遍性和灵活性,网络环境具备高保真性、多样性、安全性、可扩展性和可观测性,部署广泛并且可接入大量用户从而方便网络和应用的测试与评估,已部署的服务和原型可长期使用,以及资源隔离可控等[26].

GENI 包含 2 个关键部分:①由可扩展的标准组件组成的基底网络;②全球的控制和管理框架,它把这些标准组件装配形成一个一体的设施.在 GENI 中主要有 2 种活动:①把小型和中型的实验平台(如校园网中的 openflow 实验平台)联合起来配置成一个原型网络;②在 GENI 上运行可观测、可控制和可记录的实验.目前 GENI 由美国的一组研究项目组成,它们聚集在 4 个控制框架之下并且相互合作与竞争,即 PlanetLab,ProtoGENI[27](由 Emulab[28]派生而来),ORCA[29](open resource control architecture)和 ORBIT[30].

3) VINI

VINI(virtual network infrastructure)[31]是对 PlanetLab 的扩展,它可以支持如 X-Bone 或 VIOLIN 中的虚拟网络,是最终完成 GENI 前的一个中间产物.与 PlanetLab 相比,它同样是一种虚拟化的网络实验设施,同样允许科研人员在一个真实的环境中评价他们的协议和服务,但它的可控性更高.VINI 在路由层上给研究人员提供比 PlanetLab 更大的自由,它具备创建复杂真实网络的能力,能够将外部事件注入网络,从而为提议的网络结构创造出一种比模拟和仿真更真实的实验手段.

VINI 原型(PL-VINI)最初建立在 PlanetLab 上,它将一组现有的软件合并到 PlanetLab 上,因此可被视为一种特殊的使用软件路由器的 Overlay 网络,并且这些 Overlay 网络可以并存.具体地说,它在 PlanetLab 的 slice 上安装用户模式的 Linux(UML),使用 XORP 实现路由功能,使用 Click 实现分组转发和网络地址翻译,使用 OpenVPN 服务器来连接最终用户.然而由于 VINI 路由器运行在 PlanetLab 的 Overlay 网络上,并没有直接相连,因此实际路径上的路由器性能以及它们的路由协议会对网络层实验产生影响.PL-VINI 中的路由器构造如图 4 所示.

后来 VINI 升级到自己的物理硬件上并使用新的软件平台 Trellis[32].Trellis 使用 GRE 隧道连接不同物理机器上的虚拟节点,由于这些隧道通过一个类似以太网的接口连接虚拟节点,因此也被称为

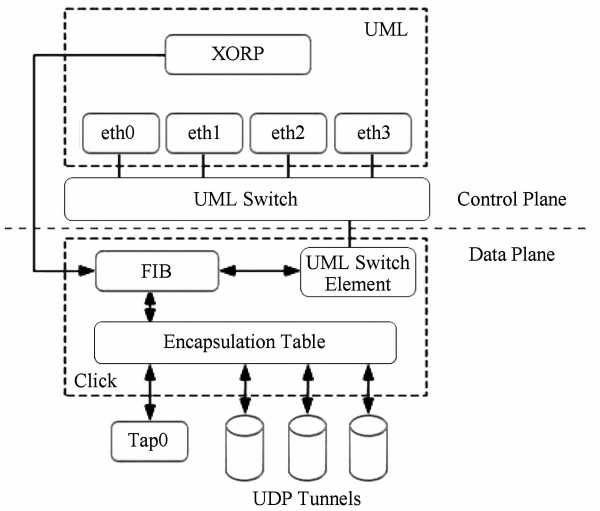

Fig. 4 The configuration of slice router in PL-VINI.

图 4 PL-VINI 中的 slice 路由器构造

以太网通用路由封装隧道(EGRE). 在每台物理机器内部,Trellis 使用 Linux 的桥(bridge)模块将一条 EGRE 隧道连接到多个虚拟接口,并使用短桥(shortbridge)优化 EGRE 隧道与虚拟接口的一对一连接.

4) FEDERICA

FEDERICA (federated e-infrastructure dedicated to European researchers innovating in computing network architectures)[33-34]是由欧洲多个组织共同资助、能够对新型互联网结构和协议进行实验的大规模网络基础设施,用于支持 FP7 下的 FIRE (future Internet research and experimentation)项目,即未来互联网开发. 它致力于在可虚拟化的网络基础设施中解决数据平面、控制平面和管理平面中的挑战,旨在提供一个泛欧、透明和协议无关(agnostic)的基底网络,支持多个 slice 共存并允许用户对尽可能低的网络层次拥有完全控制,而且多个用户 slice 可以互相连接,也可以连接到外部的网络和服务以创建更大的联合网络.

FEDERICA 的设计原则之一是强调实验的可重复性,即给定同样的初始条件,多次实验的结果保持不变. 它使用可编程的高端路由器和交换机作为核心节点,使用基于 PC 的虚拟机作为非核心节点,使用多协议交换机连接核心节点和与之对应的非核心节点. 在 FEDERICA 中,虚拟网络由一个中心的准入控制和决策过程创建,分配给不同用户的 slice 和 FEDERICA 资源由一个专门的代理保证它们不会受到未经授权的非法访问.

与流行的 PlanetLab 平台不同的是,FEDERICA 中的节点不仅可以作为终端主机,而且可以作为可编程的网络和计算设备对网络流量进行路由,并能影响网络的可达性;另外,用户不必使用指定的操作系统,可以自由地在 slice 中安装任意软件,以及可以在更低的协议栈层次上进行网络实验.

5.2 软件定义网络的虚拟化

未来互联网虚拟化的设计目标之一是可以通过编程实现新的互联网范式,对可编程网络的研究促进了 SDN 范式的出现. 尽管 SDN 被建议用于或已经应用在企业网、数据中心、基于基础设施的无线接入网、家庭和小型商业机构等场合,但它未来还将进一步发展、用于实现软件定义的互联网[35]. 本节描述当前状况的软件定义网络的虚拟化,或者对基于 OpenFlow 网络模型实现的 SDN 的虚拟化现状进行描述.

SDN 是一种新网络结构,旨在将网络的控制平面与物理拓扑解耦合. 这种网络结构的重要特征是拥有一个网络级的操作系统,能够将整个网络的逻辑拓扑展露给位于集中控制平面上的控制程序或服务. SDN 的主要思想是使软件开发者能够像依赖存储和计算资源那样简易的方式依靠网络资源实现联网(networking). 在 SDN 中,网络智能被逻辑地集中在位于控制平面并基于软件的控制器上,网络设备(通常称作交换机)只是在数据平面上进行简单地分组转发,并且控制器可以通过开放接口(例如 OpenFlow)对它们编程[36]. 通过将网络的转发硬件

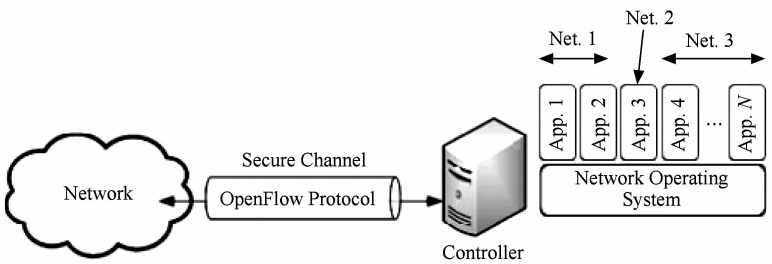

Fig. 5 The model of OpenFlow controller.

图 5 OpenFlow 控制器模型

与控制逻辑分离,SDN 有希望显著降低网络管理负担并允许网络创新和发展. SDN 控制器模型如图 5 所示[37].

1) 基本的虚拟化方法

为了允许多个用户或应用分享交换机,OpenFlow 需要经过一个对底层网络设备、链路和拓扑进行抽象的虚拟化层,即 FlowVisor[38]. 它位于 OpenFlow 控制器和 OpenFlow 交换机之间,可以对底层网络资源进行适当的切分然后分给不同的用户,而且用户通过它可以无缝地控制和编排(orchestrate)分配的虚拟资源,用于有效地交付网络服务. 在 OpenFlow 网络模型中,虚拟网络由其控制平面以及控制平面控制的流定义,它的拓扑由网络中当前存在的流决定,而且其实例化只是在控制器上创建一组用于该虚拟网络的应用程序. 通过 FlowVisor,每个虚拟网络的 OpenFlow 控制器只能控制自己所在的虚拟网络中的资源.

为了在多个虚拟网络之间进行隔离,FlowVisor 提供了链路带宽、网络拓扑、流空间、OpenFlow 交换机的 CPU、流表项以及 OpenFlow 控制这 6 种隔离机制. 例如 FlowVisor 为共享同一个交换机端口的每个虚拟网络配置一个最小带宽队列实现链路带宽隔离,通过限制每个用户可发送的控制信息的数量实现交换机的 CPU 隔离,通过限制每个虚拟网络可使用的流表条目数量实现交换机流表项的隔离,通过改写虚拟网络的控制信息实现 OpenFlow 控制隔离.

2) 面向应用的虚拟化方法

上述 1)提到的 FlowVisor 是对网络数据平面的虚拟化,并且采用的是对底层网络资源切分的方法. 这种方法基于基础设施即服务模型(IaaS),允许多个用户分享底层的物理网络资源;然而在这种模型下,用户将面临运营和管理虚拟网络的所有常规而又复杂的问题,例如处理网络拥塞、节点和链路故障以及大量网络设备的配置自动化问题. 对于需要直接和完全控制网络的用户来说,它们不成问题;然而对于专注于在虚拟网络上提供增值服务的用户来说,他们并不愿承受这种网络管理负担. 为了使基于 SDN 的网络虚拟化能够同时满足这 2 类用户的需要,Corin 等人[39]提出了一种新的网络虚拟化结构 VeRTIGO. 其基本思想是通过给 FlowVisor 增添额外的智能,可根据用户的需要将不同的网络视图展露给他们的控制器. 具体地说,VeRTIGO 可根据用户要求,既可以对由虚拟节点和虚拟链路组成的虚拟网络实例化,也可以对整个网络塌缩成的抽象节点实例化. 即用户可以在以下 2 种极端选项中选择:①完整的虚拟网络,虚拟网络拓扑完全按照用户的需要定制,并且用户能完全控制虚拟网络;②单个的抽象节点,用户只需关注路由策略,由基础设施提供者管理分布于不同地理位置的底层虚拟网络. VeRTIGO 的组成模块如图 6 所示:

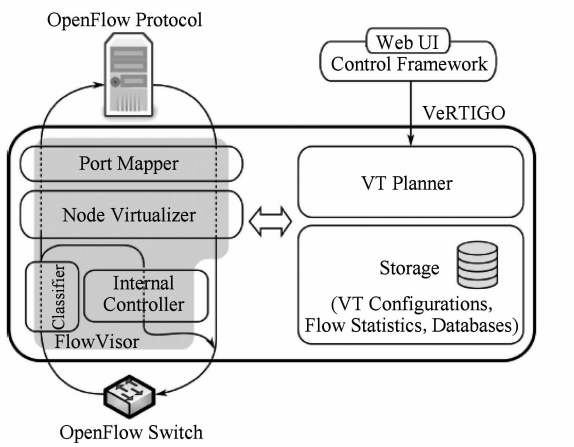

Fig. 6　The main components in VeRTIGO.

图 6　VeRTIGO 的主要构件

其中,Classifier 模块根据存储在配置文件中的信息对交换机发出的 OpenFlow 消息进行分类. 在单个抽象节点场景下,只有与接入端口有关的消息才会被发送给 OpenFlow 控制器,其余消息由 VeRTIGO 的内部控制器(internal controller)处理. 在虚拟网络场景下,只有虚拟链路的端节点才能与 OpenFlow 控制器通信,所有其他节点则被 VeRTIGO 隐藏起来,由它直接控制这些节点的流表. 节点虚拟器(node virtualizer)用于单个抽象节点的场景,它将位于 VeRTIGO 与物理网络之间的多个控制通道复用到位于 VeRTIGO 与 OpenFlow 控制器之间的一条控制通道上. 在虚拟网络场景下,由于一个物理端口可能同时终结多条虚拟链路,因此与这种物理端口有关的 OpenFlow 消息在发送给 OpenFlow 控制器之前,需要端口映射器(port mapper)对其修改以与对应的虚拟端口联系起来,此外在单个抽象节点场景下端口映射器还用于重新映射接入端口号. 内部控制器用来控制所有对 OpenFlow 控制器隐藏的交换机,它们包括单个节点场景下的所有交换机和虚拟网络场景下不是虚拟链路端点的交换机. 存储模块提供一组操作指令和函数用于储存抽象节点和虚拟网络场景下的配置信息,以及处理与流的头部和统计信息有关的数据库. 虚拟拓扑规划器(VT

planner)实现的是路径选择算法,用于有效地选择组成虚拟链路的物理节点和链路.用户接口和控制框架(UI and control framework)则用于简化基础设施提供者对虚拟网络和抽象节点的配置.

3) 可扩展的虚拟化方法

Bozakov 等人[40]也注意到基础设施提供者(或基底网络运营者)实例化虚拟 SDN 网络(vSDN)工作的复杂性,例如映射 vSDN 拓扑、为隧道(即虚拟链路)安装辅助的流表项以及实施流表隔离等,提出一种能够自动和可扩展地切分资源的 Hypervisor 设计——AutoSlice.在控制平面,他们采用分布式的 Hypervisor 结构,包含一个管理模块(MM)和多个控制器代理(CPX).通过将物理基底网络划分为多个 SDN 域,并为每个 SDN 域配备一个专用的 CPX 分担控制负载,从而使 Hypervisor 能够处理来自多个用户的大量流表控制消息.当收到用户的 vSDN 请求后,MM 将 vSDN 拓扑映射到多个 SDN 域内,并将对应的虚拟资源子集分配给每个域内 CPX,然后由各个 CPX 通过在选定的交换机上安装流表项完成 vSDN 子拓扑的实例化.

在转发平面,为了减轻 SDN 网络在多用户环境下因扩展性对 OpenFlow 交换机流表容量的压力,在每个 SDN 域内配备一个 ASD(auxiliary software datapath),ASD 是一台运行软件交换机(例如 OpenVSwitch)的服务器,具有足够的内存储存该 ASD 所承担的逻辑流的表项.根据网络流量的 Zipf 属性,可用 ASD 处理低容量的流(老鼠流),而在 OpenFlow 交换机的流表里直接缓存少量的大容量流(大象流),从而实现 SDN 网络数据平面的可扩展性.

Drutskoy 等人[41]也注意到大量具有不同虚拟网络要求的用户给 SDN 网络带来的扩展性问题,提出了一种可扩展的网络虚拟化结构 FlowN.它使 SDN 网络的每个用户可以拥有自己专用的虚拟 SDN 网络,即用户可以自行支配虚拟 SDN 网络的地址空间、网络拓扑和控制器而不受物理 SDN 网络的限制.在物理 SDN 网络中支持大量具有不同拓扑的虚拟网络需要有一种方法,能够将不同虚拟网络的流规则和流查询映射到对应的物理交换机上,以及将物理事件映射到所有受其影响的虚拟组件上,此外还要求这种映射方法与物理网络的规模无关,即允许物理网络的规模可扩展.

为了解决这些问题,FlowN 结构采用了 2 种关键的设计决策:①使用基于容器的虚拟化技术将用户控制器建立在公共的控制器平台(例如 NOX)上;②利用现代数据库技术提升虚拟网络和物理网络之间映射的可扩展性.基于容器的控制器虚拟化使 FlowN 不再对 OpenFlow 消息进行映射,而是对公共控制器平台上不同用户控制器的 API 调用进行映射,从而避免为每个用户设置一个控制器并节省了系统开销.使用数据库实现虚拟拓扑与物理网络之间的映射,不但可以避免使用定制数据结构实现网络映射的不便,而且可以利用数据库复制将 FlowN 虚拟化层分散在不同的物理服务器上,从而大大增加 FlowN 服务的用户数量.

4) OpenFlow 扩展结构的虚拟化

上面 1)~3)所述的 SDN 网络虚拟化都基于目前实现的 OpenFlow 网络规范,并且使用一个共同的数据平面,即 IPv4 转发平面.为了使 OpenFlow 网络模型能够用于运营商的广域网、支持多种不同类型的网络服务,有必要对其系统结构进行扩展,进一步增加其资源分配、资源隔离和网络虚拟化的灵活性.为此,Skoldstrom 等人[42]从系统设计角度提出了一种新的高度灵活的虚拟化模型,可以实现数据路径和控制通道的虚拟化以及它们资源的公平分配.

本文作者认为网络虚拟化的基础是存在一个翻译单元(translation unit),它在物理网络和虚拟网络之间进行标识符(identifier)映射.不管怎样或在哪实现这个翻译单元,它必须位于应用逻辑和物理硬件之间,并预见到 OpenFlow 网络中存在如图 7 所示的 5 种虚拟化模型.

图 7(a)所示的是 FlowVisor 方法,翻译单元作为协议代理位于多个控制器和多个交换机之间并被它们分享.

图 7(b)所示的翻译单元位于交换机中的 OpenFlow 实例(OpenFlow instance)中,它对不同的控制器进行区分并执行协议翻译.

在图 7(c)的交换机中存在多个 OpenFlow 实例,每个 OpenFlow 实例对应一个控制器,翻译单元在 OpenFlow 实例与快速路径转发硬件之间进行映射.

在图 7(d)中,交换机上存在多个 OpenFlow 实例和多条并行的数据路径.翻译单元的功能是将交换机端口与并行路径关联.

在图 7(e)中存在多个翻译单元,其中一个负责交换机上的虚拟化,另一个负责连接多个虚拟交换机并生成虚拟网络拓扑.

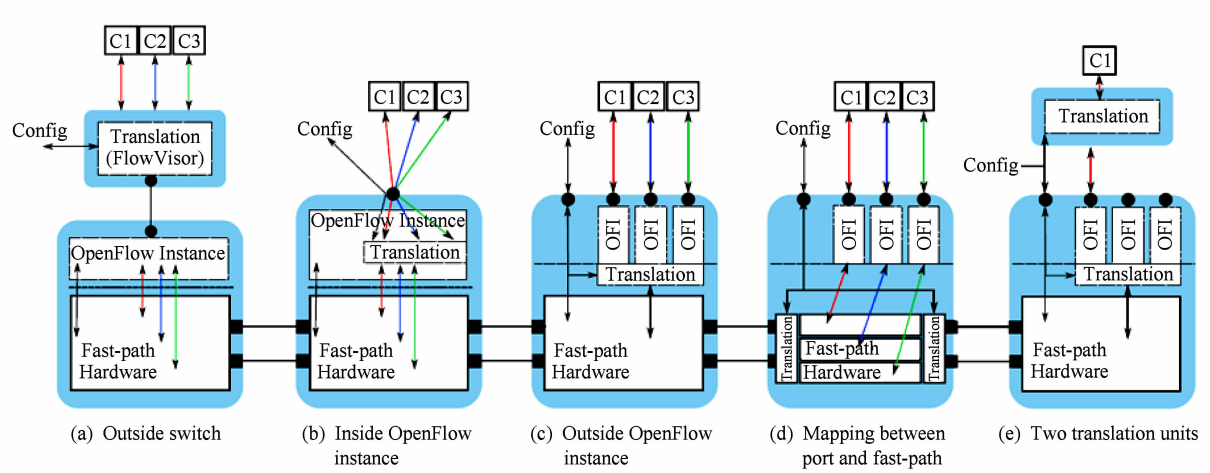

Fig. 7　The placement of translation unit.

图 7　翻译功能的放置位置示例

通过对运营商 SDN 广域网支持不同服务所需的数据路径隔离和分配、控制通道资源分配以及交换机上 OpenFlow 实例的隔离和资源分配的分析，本文提出一个功能完整的虚拟化系统如图 8 所示：

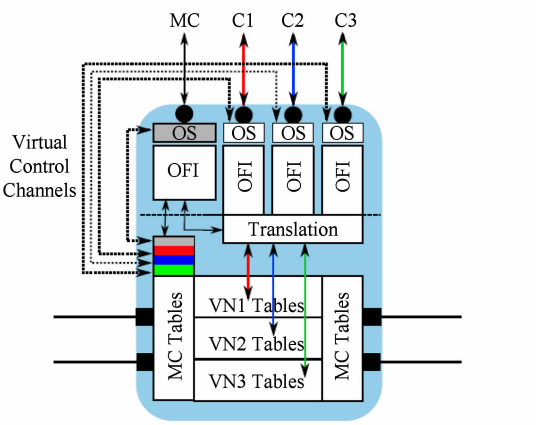

Fig. 8　The virtualization system with complete function.

图 8　功能完整的虚拟化系统

图 8 中数据路径由图 7(c)改进而来，通过在数据路径上使用专用的入口和出口流表，并利用数据封装和流表分割的方法可实现数据路径隔离. 交换机上的应用(或服务)OFI 可通过操作系统的虚拟化机制分开，并通过翻译单元连接到不同的数据路径. 对应主控制器 MC(由网络所有者操作)的 OFI 用于配置入口流表、出口流表以及翻译单元，实现虚拟网络定义以及它们与数据路径的指配. 控制网络使用带内(in band)IP 网络，并分为主控网络和虚拟控制通道 2 部分，其中虚拟控制通道由主控制器 MC 管理. 此外，控制网络的路由、QoS 以及发生故障后的重配置也由 MC 负责.

6　未来互联网虚拟化研究发展趋势

关于未来互联网的虚拟化研究，经过分析，我们认为结构多元论的观点涵盖范围更广，它不但可以把结构单元论的观点融入到未来虚拟化的互联网环境中，而且更适合人们对未来互联网可演进发展的要求.

基于这种分析，我们在表 1 中总结了未来多元结构的互联网和实验平台的虚拟化研究. 它不包括单元 clean slate 结构的虚拟化研究，然而由于单元可演进结构的虚拟化研究有可能使当前结构的互联网演进到未来多元结构的互联网，因此在表 1 中也包含了这部分的研究. 从表 1 可以发现，为了建立一个整体的、一般性的未来网络环境，互联网虚拟化研究具有以下 4 个独特的趋势：

1) 对互联网的虚拟化更加彻底. 以前的研究项目如 X-Bone，VIOLIN，PlanetLab，其研究焦点是将物理节点虚拟化然后将虚拟节点连接起来，或是在物理节点之间部署虚拟链路；而近来的研究项目如 GENI，4WARD，OpenFlow，则不仅仅实现节点和链路的虚拟化，而且试图通过有效的隔离实现网络互连其他方面的虚拟化，例如管理平面的虚拟化.

2) 未来互联网的虚拟化层次逐步降低. 例如 FEDERICA 和 CABO 就比之前同类研究的虚拟化层次低. 虚拟化发生的层次越低，未来互联网就越灵活，从而就越能实现互联网范式的多样性和丰富性. 然而这种灵活性也有其不利的一面，例如这样产生的虚拟互联网没有全网级的控制和管理平面，目前

对这个问题仍在继续研究,而且有不同的解决办法,例如设立一个中心的权威机构[43]或利用网络联盟(federation)等.

3) 未来互联网虚拟化研究趋向异构环境.以前的研究专注于利用某种特定网络技术的特征进行虚拟化,但 GENI,4WARD 和 FEDERICA 等项目近来考虑更多的是如何将不同网络技术的虚拟化方法整合在一起.

4) 基于控制平面与转发平面分离并且具有全网集中控制平面的网络结构是 SDN 的核心特征,这种联网范式能够简化网络管理、增加网络灵活性并允许网络创新,正吸引越来越多的关注并形成一种新的网络设计趋势,有望成为未来互联网的元结构(meta-architecture)[44]或其组成部分.

Table 1 The Characteristics of Researches on Future Internet

表 1 未来互联网虚拟化研究及特征

Class	Project	Purpose	Technology	Layer	Granularity
Architecture	VIOLIN	Deploy value-added service on demand in IP overlay.	IP	Application	Node
	X-Bone	IP overlay deploys automation.	IP	Network	Node/Link
	CABO	Deploy value-added service end to end on shared infrastructure.	Heterogeneous	Network	Complete
	4WARD	Virtual network instantiation, deployment and management in business environment	Heterogeneous	Network	Complete
Testbed	PlanetLab	Overlay testbed deployment and management	IP	Application	Node
	GENI	Testbed which can customize virtual network	Heterogeneous	(on-going)	(on-going)
	VINI	Protocol and service valuation in real network environment	Heterogeneous	Link	Node
	FEDERICA	Experimental infrastructure with reproducible results	Heterogeneous	Link	Node/Link
	OpenFlow	Software defined network with product and test in one body	Heterogeneous	Link	Complete

需要指出的是,所有的互联网虚拟化技术和方法并不是凭空产生或从头开始的,而是大量借鉴了过去的研究成果.例如 VPN 和 Overlay 网络中的隧道机制为创建虚拟链路提供了示范和基准,主动和可编程网络中提出的可编程网络的概念为通过虚拟化路由节点实现互联网虚拟化的方法提供了灵感,SDN 网络结构则起源于 4D[45]和 SANE/Ethane[46]项目.此外其他的一些概念如 QoS、DiffServ、IntServ、SOA、IaaS、云计算等,也都对未来互联网虚拟化设计产生影响,甚至起着指导作用.

7 总结和未来工作

传统互联网经过数十年的快速成长和发展,其自身的巨大规模和端到端的设计原则使其结构升级异常困难.为了应对互联网出现的结构僵化趋势,人们提议从头开始重新设计互联网.与重新设计一个万能的互联网结构不同的是,研究人员认为可通过网络虚拟化创建一个有多种用途的未来互联网,它足够灵活,可以在同一个基底网络上同时支持多种互联网范式,而且能够发展演进、适应未来的变化并满足未来的要求.

本文分别从互联网体系结构和实验平台 2 方面对各相关研究进行了概述和分析,并认为网络虚拟化既可以用于基底网络,即它可以用来形成一种网络环境,使多种体系结构的互联网能够并存;同时也可以用于互联网体系结构,即允许对互联网体系结构进行继承,从而为基于特定互联网结构的创新创造便利,提供一种类似结构模板的功能.在未来互联网中同时应用这 2 种类型的网络虚拟化,将使其能够可持续演进,从而满足不断出现的各种应用的要求.本文最后基于此观点,总结了未来互联网虚拟化研究的发展趋势.

尽管互联网虚拟化很有前景、可以给未来互联网提供开放性、灵活性和异构性,但要实现未来虚拟化的网络环境,还有一系列挑战问题需要解决[47],例如虚拟网络实例化、虚拟网络运营和管理等,并需要网络研究人员、分布式系统研究人员以及其他领域相关研究人员付出艰苦努力并开展广泛合作.

参 考 文 献

[1] National Research Council. Looking Over the Fence at Networks [M]. Washington: National Academy Press, 2001

[2] Peterson L, Anderson T, Culler D, et al. A blueprint for introducing disruptive technology into the Internet [J]. ACM SIGCOMM Computer Communication Review, 2003, 33(1): 59-64

[3] Anderson T, Peterson L, Shenker S, et al. Overcoming the Internet impasse through virtualization [J]. Computer, 2005, 38(4): 34-41

[4] Wang A, Iyer M, Dutta R, et al. Network virtualization: Technologies, perspectives, and frontiers [J]. Journal of Lightwave Technology, 2013, 31(4): 523-537

[5] Khan A, Zugenmaier A, Jurca D, et al. Network virtualization: A hypervisor for the Internet? [J]. Communications Magazine, 2012, 50(1): 136-143

[6] Cisco. Network virtualization-path isolation design guide [OL]. [2014-04-02]. http://www.cisco.com/en/US/docs/solutions/Enterprise/Network_Virtualization/PathIsol.html

[7] Nakao A. Network virtualization as foundation for enabling new network architectures and applications [J]. IEICE Trans on Communications, 2010, 93(3): 454-457

[8] Chowdhury N M, Boutaba R. A survey of network virtualization [J]. Computer Networks, 2010, 54(5): 862-876

[9] Clark D D, Sollins K, Wroclawski J, et al. Addressing reality: An architectural response to real-world demands on the evolving Internet [J]. ACM SIGCOMM Computer Communication Review, 2003, 33(4): 247-257

[10] Crowcroft J, Hand S, Mortier R, et al. Plutarch: An argument for network pluralism [J]. ACM SIGCOMM Computer Communication Review, 2003, 33(4): 258-266

[11] McKeown N, Anderson T, Balakrishnan H, et al. OpenFlow: Enabling innovation in campus networks [J]. ACM SIGCOMM Computer Communication Review, 2008, 38(2): 69-74

[12] Jiang X, Xu D. Violin: Virtual Internetworking on overlay infrastructure [G] //Parallel and Distributed Processing and Applications. Berlin: Springer, 2005: 937-946

[13] Touch J. Dynamic Internet overlay deployment and management using the X-Bone [J]. Computer Networks, 2001, 36(2): 117-135

[14] Touch J, Wang Y, Eggert L, et al. A virtual Internet architecture, ISI-TR-2003-570 [R]. Los Angel: Southern California University, 2003

[15] Touch J, Wang Y, Pingali V. A recursive network architecture, ISI-TR-2006-626 [R]. Los Angel: Southern California University, 2006

[16] Touch J D, Pingali V K. The RNA metaprotocol [C] //Proc of the 17th Int Conf on Computer Communications and Networks. Piscataway, NJ: IEEE, 2008: 1-6

[17] Jain R. Internet 3.0: Ten problems with current Internet architecture and solutions for the next generation [C] //Proc of Military Communications Conf. Piscataway, NJ: IEEE, 2006: 1-9

[18] Paul S, Jain R, Pan J. Multi-tier diversified architecture for the next generation Internet [C/OL] //Proc of Cloud Computing and Virtualization Conf(CCV 2010). 2010 [2014-04-02]. http://http://www.cse.wustl.edu/~jain/papers/ftp/ccv10.pdf

[19] Feamster N, Gao L, Rexford J. How to lease the Internet in your spare time [J]. ACM SIGCOMM Computer Communication Review, 2007, 37(1): 61-64

[20] Wang Y, Keller E, Biskeborn B, et al. Virtual routers on the move: Live router migration as a network-management primitive [J]. ACM SIGCOMM Computer Communication Review, 2008, 38(4): 231-242

[21] Keller E, Lee R B, Rexford J. Accountability in hosted virtual networks [C] //Proc of the 1st ACM Workshop on Virtualized Infrastructure Systems and Architectures. New York: ACM, 2009: 29-36

[22] Zhu Y, Bavier A, Feamster N, et al. UFO: A resilient layered routing architecture [J]. ACM SIGCOMM Computer Communication Review, 2008, 38(5): 59-62

[23] Correia L, Abramowicz H, Johnsson M, et al. Architecture and Design for the Future Internet: 4WARD Project [M]. Berlin: Springer, 2011

[24] PlanetLab [OL]. [2014-04-02]. http://www.planet-lab.org

[25] The global environment for network innovations (GENI) [OL]. [2014-04-02]. http://www.geni.net

[26] Peterson L, Anderson T, Blumenthal D, et al. GENI design principles [J]. Computer, 2006, 39(9): 102-105

[27] ProtoGENI [OL]. [2014-04-02]. http://www.protogeni.net

[28] White B, Lepreau J, Stoller L, et al. An integrated experimental environment for distributed systems and networks [J]. ACM SIGOPS Operating Systems Review, 2002, 36(SI): 255-270

[29] Chase J, Grit L, Irwin D, et al. Beyond virtual data centers: Toward an open resource control architecture [C] //Proc of Selected Papers from the Int Conf on the Virtual Computing Initiative. New York: ACM, 2007

[30] Ott M, Seskar I, Siraccusa R, et al. ORBIT testbed software architecture: Supporting experiments as a service [C] //Proc of the 1st Int Conf on Testbeds and Research Infrastructures for the Development Networks and Communities (TRIDENTCOM'05). Piscataway, NJ: IEEE, 2005: 136-145

[31] Bavier A, Feamster N, Huang M, et al. In VINI veritas: Realistic and controlled network experimentation [J]. ACM SIGCOMM Computer Communication Review, 2006, 36(4): 3-14

[32] Bhatia S, Motiwala M, Muhlbauer W, et al. Trellis: A platform for building flexible, fast virtual networks on commodity hardware [C] //Proc of the 2008 ACM CONEXT. New York: ACM, 2008: 1-6

[33] Szegedi P, Figuerola S, Campanella M, et al. With evolution for revolution: The FEDERICA approach [J]. IEEE Communications Magazine, 2009, 47(7): 34-39

[34] Campanella M. Federated e-infrastructure dedicated to European researchers innovating in computing network architectures [C/OL] //Proc of Future Internet Conf. 2008 [2014-04-02]. http://www.fp7-federica.eu

[35] Nunes B, Mendonca M, Nguyen X, et al. A survey of software-defined networking: Past, present, and future of programmable networks [J]. IEEE Communications Surveys and Tutorials, 2014, 16(3): 1617-1634

[36] Limoncelli T A. OpenFlow: A radical new idea in networking [J]. Communications of the ACM, 2012, 55(8): 42-47

[37] Sherwood R, Gibb G, Yap K-K, et al. Can the production network be the testbed? [C] //Proc of the 9th USENIX Conf on Operating Systems Design and Implementation. Berkeley, CA: USENIX Association, 2010: 1-6

[38] Sherwood R, Gibb G, Yap K K, et al. FlowVisor: A network virtualization layer, OpenFlow-tr-2009-1 [R]. Palo Alto, California: Stanford University, 2009

[39] Corin R D, Gerola M, Riggio R, et al. VeRTIGO: Network virtualization and beyond [C] //Proc of the 2012 European Workshop on Software Defined Networking (EWSDN). Los Alamitos, CA: IEEE Computer Society, 2012: 24-29

[40] Bozakov Z, Papadimitriou P. Autoslice: Automated and scalable slicing for software-defined networks [C] //Proc of the 2012 ACM Conf on CoNEXT Student Workshop. New York: ACM, 2012: 3-4

[41] Drutskoy D, Keller E, Rexford J. Scalable network virtualization in software-defined networks [J]. Internet Computing, 2013, 17(2): 20-27

[42] Skoldstrom P, Yedavalli K. Network virtualization and resource allocation in OpenFlow-based wide area networks [C] //Proc of 2012 IEEE Int Conf on Communications (ICC). Piscataway, NJ: IEEE, 2012: 6622-6626

[43] Caesar M, Caldwell D, Feamster N, et al. Design and implementation of a routing control platform [C] //Proc of the 2nd Conf on Symp on Networked Systems Design & Implementation, Vol 2. Berkeley, CA: USENIX Association, 2005: 15-28

[44] Niebert N, El Khayat I, Baucke S, et al. Network virtualization: A viable path towards the future Internet [J]. Wireless Personal Communications, 2008, 45(4): 511-520

[45] Greenberg A, Hjalmtysson G, Maltz D, et al. A clean slate 4D approach to network control and management [J]. ACM SIGCOMM Computer Communication Review, 2005, 35(5): 41-54

[46] Casado M, Freedman M, Pettit J, et al. Ethane: Taking control of the enterprise [J]. ACM SIGCOMM Computer Communication Review, 2007, 37(4): 1-12

[47] Li Xiaoling, Guo Changguo, Li Xiaoyong, et al. A constraint optimization based mapping method for virtual network [J]. Journal of Computer Research and Development, 2012, 49(8): 1601-1610 (in Chinese)
(李小玲, 郭长国, 李小勇, 等. 一种基于约束优化的虚拟网络映射方法[J]. 计算机研究与发展, 2012, 49(8): 1601-1610)

Yu Tao, born in 1973. PhD from Tsinghua University. His main research interests include network virtualization, computer network architecture and future Internet (yutao73@gmail.com).

Bi Jun, born in 1972. Professor and PhD supervisor at Tsinghua University. Senior member of IEEE, ACM and China Computer Federation. His main research interests include computer network architecture and next generation Internet.

Wu Jianping, born in 1953. Professor and PhD supervisor at Tsinghua University. Senior member of China Computer Federation and IEEE. His main research interests include computer network architecture and next generation Internet (jianping@cernet.edu.cn).