



面向租户的虚拟机定制化组网技术

庄子睿,王敬宇,徐 童

(北京邮电大学网络与交换技术国家重点实验室 北京 100876)

摘 要:基于 OpenStack 设计并实现了一个面向租户的虚拟机定制化组网系统,通过资源的元数据定义和基于模板的协同操作,实现云基础设施(计算、存储和网络资源)的自动化部署和管控。具体功能包括统一的图形化租户可拖拽操作、定制化的虚拟机组网编排、动态批量的资源配置更新、网络接口粒度上的服务质量控制以及基于虚拟机网络连接关系的优化映射算法等。该系统可以为租户提供网络配置能力,使租户可以方便、快捷而不受制于技术细节地进行虚拟机组网;性能上降低了网络平均时延,实现了资源的高效匹配,提高了资源利用率。

关键词:基础设施虚拟化;定制化组网;可视化云服务;OpenStack 资源编排

doi: 10.11959/j.issn.1000-0801.2015267

Tenant-Oriented Customized VM Networking Technology

Zhuang Zirui, Wang Jingyu, Xu Tong

(State Key Laboratory of Networking and Switching Technology, Beijing University of Posts and Telecommunications, Beijing 100876, China)

Abstract: A tenant-oriented customized virtual machine networking system was designed and implemented based on OpenStack. By utilizing metadata definition and template based collaboration, the system implemented automated deployment and control of cloud infrastructures. Features such as unified graphical and draggable tenant interfaces, customized virtual machine networking orchestration, dynamic and batch resource configuration, quality of service control on network interfaces, and a connectivity based optimized mapping algorithm were included. The system provided users ability to configure networks, so that users can manage to conduct virtual machine networking swiftly and without knowledge of underlying technical details. It contributed to reduction of overall virtual networking latency, high efficiency matching of resources and increase in resource utilization.

Key words: infrastructure virtualization, virtualized networking, visualized cloud service, OpenStack orchestration

收稿日期:2015-06-19;修回日期:2015-10-10

基金项目:国家重点基础研究发展计划(“973”计划)基金资助项目(No.2013CB329102),国家自然科学基金资助项目(No.61471063, No. 61421061, No. 61372120, No. 61271019),教育部科学技术研究重点(重大)项目(No.MCM20130310),北京市自然科学基金资助项目(No.4152039),北京高等学校青年英才计划项目(No.YETP0473)

Foundation Items: The National Key Basic Research Program of China(973 Program)(No. 2013CB329102), The National Natural Science Foundation of China (No. 61471063, No.61421061, No.61372120, No.61302087, No.61271019),The Key (Keygrant) Project of Chinese Ministry of Education (No. MCM20130310),Beijing Natural Science Foundation (No. 4152039), Beijing Higher Education Young Elite Teacher Project (No. YETP0473)

论文引用格式:庄子睿,王敬宇,徐童.面向租户的虚拟机定制化组网技术.电信科学,2015267

Zhuang Z R, Wang J Y, Xu T. Tenant-oriented customized VM networking technology. Telecommunications Science, 2015267



1 引言

近年来,随着云计算产业的蓬勃发展,电信运营商、信息技术企业、互联网公司的不断投入,云服务种类快速增加、功能逐渐完善。各式各样的云服务从各种各样的角度简化了租户的部署、维护和管理难度,也在相当程度上减轻了租户的资本性支出负担和运营成本。愈发成熟的技术特征和保障也吸引了越来越多的租户,从传统的信息通信技术产业领域,电信行业、互联网行业,扩展到物流、制造、教育、公共服务等各行各业^[1],乃至个人最终租户。租户接触、使用信息技术的方式也从最原始的自行购买、组建、维护和管理模式,转变成向云提供商购买服务的模式,方便、快捷、省心省力。

租户需要在各种物理设备之间迁移数据和应用,搭建云计算平台的厂商必须提供按需伸缩容量的能力,让租户根据需要很容易地配置新的服务器和存储设备。因而,在虚拟化的基础上,需要一个优化的体系结构对异构的软硬件资源进行互联,高可扩展性地增加、删除资源,并根据租户的需求自动伸缩,充分利用虚拟化的好处,方便租户在虚拟机上直接部署所定制的网络环境。

从云计算的服务层次来讲,网络服务常常被视为底层的基础设施,仅提供满足基本需求的网络通信能力,而没有给租户开放定制网络关系、进行组网规划的能力,难以满足特定租户多样化、定制化的灵活组网需求。目前在云计算环境中,云网络模型通常只把基础的动态或者静态地址以及基本的防火墙等功能分配给租户所使用的虚拟机。网络关键服务的缺乏极大地限制了租户的使用、网络的安全和性能,那么如何帮助实现灵活、可扩展地设计和部署网络服务——这正是 NaaS(network as a service,网络即服务)提供的功能。

NaaS 是用来支持应用的虚拟网络,与连接固定地点(如总部)到分支机构、服务器到服务器以及服务器到存储等的传统网络相比,有明显的区别。如果 NaaS 是应用网络,一个重要问题是它如何关联到网络基础设施和网络服务。这个问题的答案是将 NaaS 从美好的想法转变成“真正的 NaaS”的关键。

因此,本文专注研究基于数据中心的虚拟机定制化组网技术,重点在于在数据中心的环境下为租户提供虚拟机间灵活的网络关系定制,使租户能遵照自己应用的具体需求,以相对简单易行的操作、低廉的学习和部署成本对虚

拟机进行组网编排(orchestration)。这一编排技术的应用,对于企业租户来说,可以根据自己企业内部的组织架构或者功能划分,从网络基础设施组建的层次上进行多维度的网络访问控制和均衡,并且可以对应企业内部的变化进行敏捷快速的调整;对于科研租户来说,则可以利用灵活开放的网络基础设施,搭建符合科研需求的实验环境,能够在准真实环境下对研究成果进行验证和测试。

已有的研究诸如覆盖网络仿真软件 OverSim^[2],使用了远程过程调用进行分布式系统组件之间的沟通^[3,4],使用了管理器模式组织应用程序接口,有一定的灵活性^[5],但是其功能也受到限制,仅针对覆盖网络提供测试仿真能力。与先前的研究^[6]相比,本定制化组网系统基于数据中心场景提供了面向租户开放,针对计算和网络资源的编排能力和操作;同时关注编排实施过程中,平台边缘侧虚拟机的智能化协同生成和放置,提供了优化的虚拟资源部署。本系统不仅仅基于处理器、内存等传统资源的利用状况^[7],还考虑了虚拟基础设施的网络连接关系、网络性能需求等,涉及一系列资源的组网优化。为此,本系统采用分而治之的方法,向上结合租户编排需求,向下探测基础设施状态,将大问题化成若干小问题,对解决 NP 难问题提出了可行的解决方案。

2 编排系统

资源编排系统负责解析、处理组网需求数据,并将处理好的需求数据转化为基础设施平台可以识别的任务。该系统需要能够检测输入数据的完整性、经受不合法的输入数据的干扰,能够智能化地分析租户所请求的虚拟基础设施资源之间的依赖关系、合理地安排相关资源的生成。除此之外,还应当对原有基础设施平台的所有配置提供兼容性支持。

在数据输入部分,基础设施描述需要包含云主机、交换机、路由器、负载均衡器、防火墙,周边支持资源描述还包含密钥对。数据输入需要使用一种半结构化的、方便扩展的数据格式进行封装。

资源处理部分需要能够灵活地识别租户在平台上已经拥有的资源、新请求的资源,并且能够以一个统一的模型表示上述各项差异化的资源信息,同时易于维护资源之间的依赖关系、方便在运行时动态地解析和修改,并且为动态调整和增量更新提供便利;在此基础上,还要能够针对租户比较模糊的组网需求进行处理,比如资源调度位置偏好等。

虚拟资源生成部分需要能够将处理好的上述各种资源需求数据,完整地转化为底层基础设施平台能够识别与理解的配置数据,同时有针对性地准备资源监测配置,提交给平台进行生成,监测并返回任务状态。

2.1 元数据定义

编排系统使用元数据(metadata)对租户所请求的资源进行定义。考虑到扩展性要求,使用半结构化的 JSON (JavaScript object notation)格式进行封装。元数据涵盖的简要内容见表 1。

表 1 元数据定义

资源类型	属性
虚拟机	名称、数量、镜像、配置、网络连接、密钥对
交换机	名称、子网配置
路由器	名称、网络连接、网关
负载均衡器	名称、网络连接、协议、均衡方法、监测、资源池、虚拟地址
防火墙	名称、策略、规则列表
密钥对	名称、公钥、私钥存储

2.2 资源表示

由于基础设施平台在具体实现时,虚拟机对虚拟子网存在依赖性,然而虚拟机描述过程中只对虚拟网络有依赖需求,虚拟子网依赖于虚拟网络,反之不然,也就是说虚拟机对虚拟子网无法从逻辑上归纳出依赖性。因此这里采取重载资源描述 HeatOrchestrationTemplate 类的 toDictionary()函数,二次检查和虚拟机资源相关的虚拟网络资源,并搜寻依赖于相应的虚拟网络资源的所有虚拟子网资源,并将它们添加到虚拟机资源的依赖项中。

出于可维护性的考虑,在进行资源描述表示时使用统一的基础类型进行表达;又由于系统需要考虑已有资源和新增资源之间在发生属性依赖时的兼容性,这里专门做出处理。特别地,在 BaseResource 类中,设置资源属性时候做出检测,如果目标赋值为 Resource 实例,则创建一个对该 Resource 实例的引用 ResourceReference 对象,并进行具体赋值。类似地,针对列表和字典内容,也有相应的解码赋值操作。

综上所述,资源统一使用 BaseResource 类的细分派生成员来表示,并且使用 ResourceReference 对象类型标识资源间的引用和依赖关系,便于统一的设计以及运行时的自动处理。

2.3 资源调度策略

虚拟机作为最小计算资源单元,是虚拟基础设施资源

的管理重点。而在虚拟基础设施资源的生成过程中,虚拟机映射位置的选择必然影响网络连接关系的映射、网络资源的安排和配置。首先,原有平台的管理模块基于传统思路,针对虚拟机的调度生成,仅考虑了诸如处理器核心数量、随机存取存储器大小、块设备大小等传统计算资源,本着筛选可用、选取最适的原则执行虚拟机的调度任务。这样的调度逻辑基本满足传统的 IaaS (infrastructure as a service,基础设施即服务)需求,但是难以满足租户在定制化的虚拟机组网下,对虚拟机间和虚拟网络内通信的优化需求。其次,原有的调度仅针对单一虚拟机资源,难以从整体角度给出优化方案。另外,租户可能需要指定若干服务节点或者服务节点集合来生成某些具体的虚拟机,这在原有平台的模式中也是不能被满足的。

基于上述各项原因,本文扩展了原虚拟基础设施平台对虚拟机的调度功能,对原有的基于处理器核心等计算能力的调度算法进行了扩展。扩展后的调度算法,依然遵循“先筛选可用,再选取最适”的思路^{[8][9]}。首先,在原有算法的基础上,根据生成位置的定义,筛选可用的服务节点;其次,将虚拟机的网络连接关系纳入考虑范围,评价各个虚拟机在各个虚拟网络内到邻近虚拟机的通信时延,通过选取最短时延的方式产生最适节点。各虚拟机对应的最适节点以及相关的资源占用将被记录下来供进一步优化调度。最终调度算法能够针对所提交的一组虚拟机调度请求,返回整体优化过的调度结果。

(1)调度消息交互序列

鉴于原有基础设施平台没有很好地考虑网络连接关系和服务节点位置对虚拟机组网通信的影响,这里对调度模块进行了扩展。外部应用通过 WSGI 提供的接口进行调用,管理器通过远程过程调用数据中间件的方法,数据中间件进一步使用远程过程调用调度器的 arrange_instances 方法,如图 1 所示。

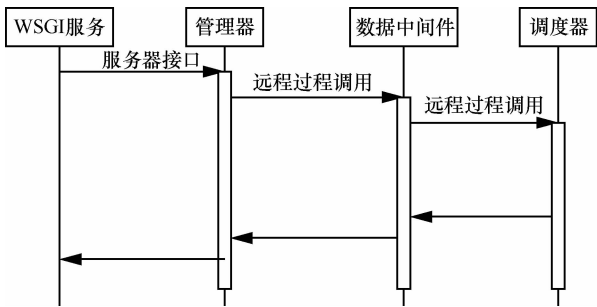


图 1 调度功能的交互序列



(2)调度工作流程

规划调度时依然采用“先过滤可用,再选取最适”的思路。首先根据组网需求中对位置的偏好过滤出一部分可用服务节点,再根据对处理器、内存、磁盘的需求,过滤出合适的可用节点。使用请求的计算能力和计算容量对节点进行加权并排序。在此基础上,考虑网络的连接关系,针对虚拟机所连接的每一个虚拟网络,计算各个可用服务节点被选中后产生的与其他虚拟机通信的平均时延。将虚拟网络选项遍历完成之后,计算各个可用服务节点针对各个虚拟网络的平均通信时延。最后对这一结果进行加权、排序,选取最优的候选作为调度结果返回。上述过程如图 2 所示。

(3)虚拟机状态维护

由于对虚拟机的调度纳入了网络连接关系作为参考依据,所以在调度请求中需要增加相应的网络连接信息。同时,出于整体优化的考虑,还需要综合考虑现有虚拟机在各个虚拟网络中的分布状态,这就需要对虚拟机状态和相关的网络连接参数进行专门的维护。

鉴于上述原因,新建立了 InstanceManager 类来负责管理、维护相关数据,可以提供虚拟机状态、虚拟机网络连接信息、指定虚拟网络中虚拟机在各个服务节点上的分布状态,并提供接口获取各个服务节点之间物理网络的性能参数。该类实例与调度器运行在同一进程,并且跟随调度器进行初始化。

2.4 映射算法

虚拟机间的虚拟网络连接关系复杂多样,虚拟网络通

信以覆盖网络的方式承载于物理链路上,利用 GRE 隧道^[10]进行连接,虚拟网络的性能很大程度上取决于所选用的物理网络。如图 3 所示,虚拟机到物理节点的映射关系实质上决定了虚拟机之间复杂的虚拟网络连接关系到物理通信链路之间的映射关系,要优化虚拟网络内的通信,则需要将虚拟网络连接关系和物理链路状态纳入虚拟机映射的算法当中。

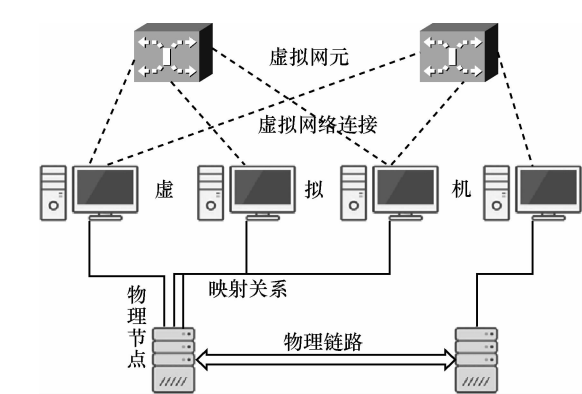


图 3 虚拟机、虚拟网络、物理节点和物理链路的关系

对于给定的一组虚拟机以及它们之间的虚拟网络连接关系,结合物理链路状态信息,可以给出更优化的映射及调度方案。如图 4 所示,给定 4 台虚拟机之间的连接关系,可以简单地将虚拟机逐一随机映射到一台可用的物理节点上,未考虑物理节点之间的通信链路状态时,方案 A 得到的平均时延是 22.5 s。分析连接关系,发现三角形内的 3 台虚拟机关联度较高;方案 B 考虑了物理链路信息,将它

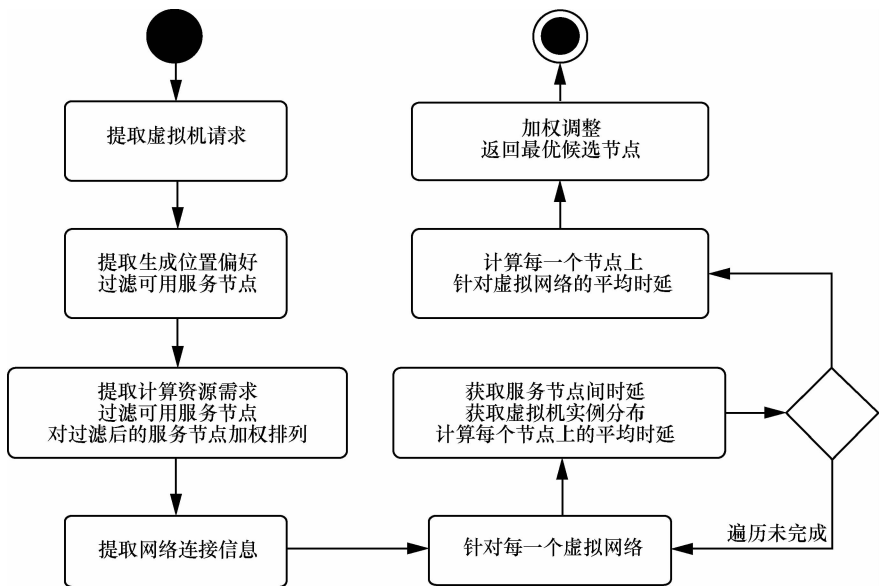


图 2 调度流程

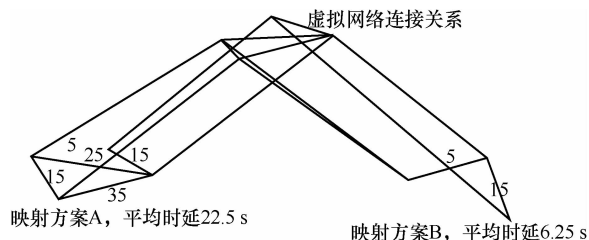


图4 随机映射方案和优化映射方案

们映射到状态比较好的两个物理节点上,剩余的一台虚拟机映射到次优的节点上,总体平均时延降到 6.25 s。

虚拟机间的通信条件受制于覆盖网络下物理节点之间的链路状态(带宽、时延等)。除了上述单一的虚拟网络,还会存在虚拟机属于多个不同的虚拟网络、分属于多个不同的物理计算节点的情景。此时应当评价待选节点在各个虚拟网络拓扑下的平均网络时延,并计算多个虚拟网络内的加权平均值。在一定的差异范围内,选取最优的若干个物理节点作为调度结果顺序返回。参见下列伪代码内容。

全局调度(上下文, 实例集):

初始化调度目的节点信息

对实例根据 `cpus, ram, disk` 进行排序

更新实例管理器信息

for 实例 in 实例集:

 获取缓存的服务节点状态

 目的节点 := 试调度(上下文, 实例, 节点集)

 更新调度目的节点信息

return 调度结果

连接关系计算判定(实例, 权重, 实例管理器):

if 权重列表为空:

return 无

if 已知虚拟机列表为空:

return 传统算法结果

总时延 := 空字典

for 虚拟网络 in 虚拟机的连接信息:

 获取当前网络映射

if 网络映射为空:

continue 处理下一个网络

 生成待选服务节点信息

 获取已有虚拟机信息

 获取节点间时延信息

 计算加权时延

计算平均时延

更新总体时延信息

if 总时延存在结果:

return 最优节点

else:

return 传统算法结果

另外,如图5所示,物理节点之间可能存在多条链路。此时基于上述思路还可以继续扩展,在对虚拟机进行映射之后,结合软件定义网络的灵活特性,可以进一步在虚拟网络链路和物理网络链路之间建立映射关系,便于对虚拟网络通信流量进行分流、隔离和服务质量控制。

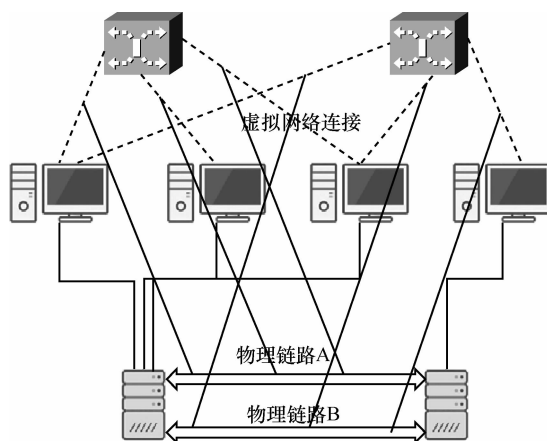


图5 虚拟网络连接到物理链路的映射

3 实验与结果分析

3.1 实验环境

实验中使用两台物理服务器节点,节点A拥有4处理核心、8 GB内存和500 GB磁盘空间,节点B拥有16处理核心、32 GB内存和500 GB磁盘空间,节点A、B之间使用100 Mbit/s网络连接。3台虚拟机配置为单处理核心、512 MB内存、10 GB磁盘空间,使用Ubuntu Server 14.04操作系统。

3.2 功能测试

在最终的实现中,通过统一的、简化的图形化租户界面入口,使用与物理设备对应的熟悉的虚拟主机、虚拟交换机、虚拟路由器、虚拟负载均衡器和虚拟防火墙的资源组织,屏蔽了原有平台的实例、网络、子网、路由、浮动IP地址、固定IP地址等专门技术细节,有利于租户使用在现实世界里先验的知识快速上手使用,完成了以可视化方式实现包括隔离网络、虚拟机、负载均衡器和防火墙在内的各种虚拟资源的生成。虚拟机的网络连接如图6所示。

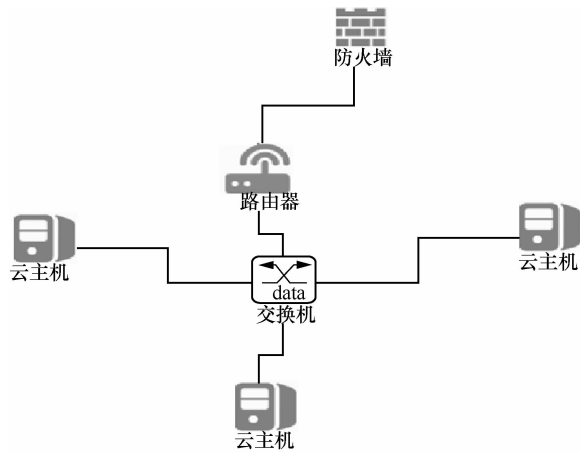


图6 图形化虚拟机组网编排系统

通过编排模块的设计,可以使用统一而简化的方式动态更改乃至批量更改资源的配置。通过对平台底层虚拟化接口的修改,实现了针对虚拟机的、多个网络接口单独定义粒度上的网络服务质量控制。

3.3 性能测试

从服务节点和虚拟机之间的虚拟网络连接关系的角度出发,对租户定制以及资源优化配置也做了一定的探索工作。由表2和表3可知,改进的映射调度算法对虚拟机间通信质量的提升有一定作用。新算法对最小RTT(round trip time)影响不大,但是对平均RTT有8%左右的提升,对最大RTT有80%左右的提升,同时可以看到,新算法调度下的虚拟机间通信有着更小的标准差,也就是更加一致且容易预测的通信表现,见表4。

表2 原有映射调度算法的虚拟机间通信

	最小 RTT/ms	平均 RTT/ms	最大 RTT/ms	标准差
A 到 B	0.716	0.933	7.866	0.713
B 到 C	0.619	0.937	12.098	1.125
C 到 A	0.733	0.913	7.641	0.683

表3 改进的映射调度算法的虚拟机间通信

	最小 RTT/ms	平均 RTT/ms	最大 RTT/ms	标准差
A 到 B	0.669	0.821	1.572	0.095
B 到 C	0.687	0.869	1.542	0.120
C 到 A	0.644	0.840	1.782	0.125

表4 改进后的算法相对于原有算法的改善

	最小 RTT	平均 RTT	最大 RTT	标准差
A 到 B	-7.57%	-12.00%	-80.02%	-0.618
B 到 C	+10.98%	-7.26%	-87.25%	-1.005
C 到 A	-12.14%	-7.99%	-76.67%	-0.558

4 结束语

随着云服务产业的成熟,越来越多的租户尝试了解、接纳甚至拥抱云服务,但云服务仍存在着显著的不足,如配置繁琐、使用门槛高、组网方式与传统实际物理机组网络模式相比有较大差别等。产业界和学术界都开始关注网络服务,本文以开源云计算管理平台 OpenStack 为基础,构建了一个虚拟机定制化组网系统,最大程度地简化租户操作,使租户可以方便、快捷、不过多地受技术细节干扰地达成预期的组网目的。

跨数据中心地域、网域的传统部署方式是分区域隔离的,这限制了资源的有效利用。未来工作将尝试用覆盖网络技术突破这一限制。跨数据中心网域、地域的系统部署,有利于服务提供商通过高效而广域的资源映射减小自身的运营成本和线路负担;能够提升平台部署的可扩展性,使得平台有可能以成机房、成数据中心的方式提升计算能力,有针对性化解资源瓶颈;能够为业务设计提供更强的灵活性,比如跨地域的虚拟设施编排、组建以及迁移,动态地跟随租户的接入位置进行优化配置生成,提供更加智能化的服务。这个方向尚存在着显著的难点需要攻克,还有很多内容需要思考与探索。

参考文献

- 1 虚拟化与云计算小组.云计算实践之道:战略蓝图与技术架构.北京:电子工业出版社,2011
Group of Virtualization and Cloud Computing. A Practical Guide to Cloud Computing: Strategy Blueprint and Technical Architecture. Beijing: Publishing House of Electronics Industry, 2011
- 2 Baumgart I, Heep B, Krause S. OverSim: a flexible overlay network simulation framework. Proceedings of the Conference on IEEE Global Internet Symposium, Anchorage, AK, 2007: 79~84
- 3 Birrell A D, Nelson B J. ACM transactions on implementing remote procedure calls. Computer Systems, 1984, 2(1)
- 4 Vinoski S. Convenience over correctness. IEEE Internet Computing, 2008, 12(4)
- 5 Lämmel R, Peyton Jones S. Scrap your boilerplate: a practical design pattern for generic programming. ACM, 2003, 38(3)
- 6 Mayoral A, Vilalta R, Muñoz R, et al. Integrated IT and network orchestration using OpenStack, OpenDaylight and active stateful PCE for intra and inter data center connectivity. Proceedings of 2014 European Conference on Optical Communication (ECOC), Cannes, France, 2014: 1~3
- 7 Sotomayor B, Montero R S, Llorente I M, et al. Virtual infrastructure management in private and hybrid clouds, IEEE

- Internet Computing, 2009, 13(5): 14~22
- 8 Hu J H, Gu J H, Sun G F, *et al.* A scheduling strategy on load balancing of virtual machine resources in cloud computing environment. Proceedings of 2010 Third International Symposium on Parallel Architectures, Algorithms and Programming (PAAP), Dalian, China, 2010: 89~96
 - 9 Sotomayor B, Montero R S, Llorente I M, *et al.* Virtual infrastructure management in private and hybrid clouds. IEEE Internet Computing, 2009, 13(5): 14~22
 - 10 IETF Network Working Group. Generic Routing Encapsulation (GRE). RFC2784, 2000

[作者简介]



庄子睿,男,北京邮电大学硕士生,主要研究方向为云计算、网络虚拟化、软件定义网络。



王敬宇,男,博士,北京邮电大学副教授、博士生导师,首批北京高等学校“青年英才计划”、IEEE 及中国通信学会会员,主要研究方向为 SDN 及未来网络、多路径传输、流量控制、云计算与虚拟化技术、大数据支撑技术等。国际上首批从事多路径传输研究的探索者,推动了流量工程在多路径领域的纵深发展。发表 SCI 检索学术论文 30 余篇,申请国家发明专利 10 余项;国家“973”计划课题负责人,主持多项国家及北京市自然科学基金项目,作为主研人参与国家及省部级重大科研课题 10 余项。



徐童,男,博士,北京邮电大学讲师,主要研究方向为业务网络智能化、移动互联网、普适服务。