Vertical Scaling Capability of OpenStack

Survey of Guest Operating Systems, Hypervisors, and the Cloud Management Platform

Marian Turowski and Alexander Lenk

FZI Forschungszentrum Informatik, Friedrichstr. 60, 10117 Berlin, Germany
{Turowski,Lenk}@fzi.de

Abstract. With the emergence of cloud computing, resources can be dynamically scaled. A common scaling approach is the addition and removal of virtual machines, known as horizontal scaling. Horizontal scaling can take several minutes but erratic and sudden changes in demand take place within seconds. Therefore, vertical scaling has been introduced, changing the resources of an existing virtual machine during run time and within one second or less. At the same time, more and more private clouds and cloud providers apply the open-source platform OpenStack. Hence, this paper evaluates the vertical scaling capability of OpenStack. For this purpose, we examine whether and to what extent common guest operating systems, popular hypervisors, and OpenStack itself support vertical scaling. Altogether, the considered operating systems and hypervisors support vertical scaling of almost all considered resources while OpenStack does not support vertical scaling at all. Based on our findings, we finally suggest steps to improve OpenStack.

Keywords: cloud computing, scalability, operating system, hypervisor, OpenStack, vertical scaling

1 Introduction and Motivation

Through the increasing popularity of the internet, the scientific community but also more and more small and medium enterprises benefit from offering their services as web applications. These web applications face a volatile demand due to scheduling, seasonal changes, temporary trends, and momentary effects [11]. Examples include experiments, time-of-day patterns as well as breaking news, marketing activities, and posts on social media such as Twitter [21]. As a consequence, applications' demand for resources is unstable and hardly predictable. In the past, the common strategy to handle the uncertain resource demand was to provide enough resources for peak workloads. This overprovision led to mostly idle resources and thus high expenses [10].

With the emergence of cloud computing, resource provision strategies changed fundamentally. Cloud computing enables on-demand provisioning of resources for experimentation [29], disaster recovery [35,36], scaling, and so on. This flexible provision and release of resources based on customer demand is an essential feature of cloud computing and known as elasticity [41]. Together with the pay-per-use billing model, clouds allow customers to pay only for the resources they are actually using and hereby to avoid over- and underprovisioning [11]. Nevertheless, adding and removing resources according to the actual demand while minimizing cost and maintaining a constant service level is a crucial issue in cloud computing [37]. A common approach for adequate scaling of resources in literature [12,18,62] and practice [8,24] is the addition and removal of virtual machines (VMs), known as horizontal scaling.

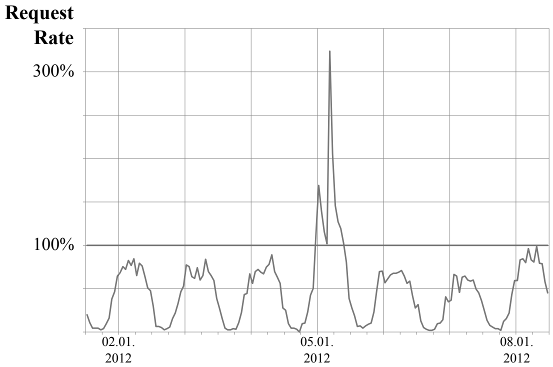

Fig. 1. Peak in the request rate of gloveler.de after the release of a newspaper article and its dissemination in social media

However, horizontal scaling comes with flaws. In scientific computing Galante et al. [23] point out that the scaling size of a whole virtual machine is not sufficient and fine-grained scaling techniques are necessary. Also erratic and sudden changes in demand take place within seconds [21] (see Fig. 1) but provisioning an additional VM requires about 50 to 800 seconds and starting multiple VMs at once may take longer than starting a single VM [39]. In these situations, horizontal scaling could lead to a decreased throughput [19], an increased response time, or even the outage of the whole system [28]. As a consequence, horizontal scaling is more appropriate for situations where changes in demand take place gradually or can be foreseen. Furthermore, additional VMs could lead to additional licensing costs [68]. For each VM, an operating system is required, which limits the additional capacity usable by the application [49]. Since VMs function as scaling unit, horizontal scaling can also cause unnecessary overprovisioning [19] and high resource wastage in case of a slight workload increase [68]. In Figure 2, such a situation is visualized, where VMs are horizontally scaled because of a demand peak. Lastly, horizontal scaling requires a running load balancer and is limited to stateless applications [19].

In order to avoid these shortcomings, vertical scaling has been introduced as a new approach. In contrast to horizontal scaling, it changes the resources of an existing VM, such as CPU cores or memory, during run time [49] (see Fig. 3). Additionally, vertical scaling can be done within one second or less [67,22]. Hereby, vertical scaling reduces violations of a certain service level [19].

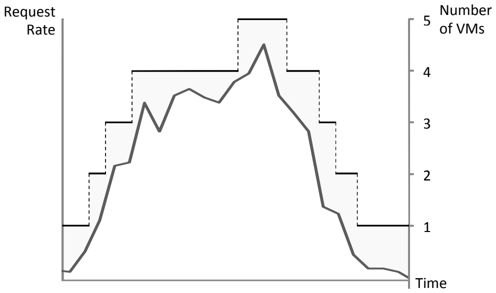

Fig. 2. Overprovision and resource wastage with horizontal scaling

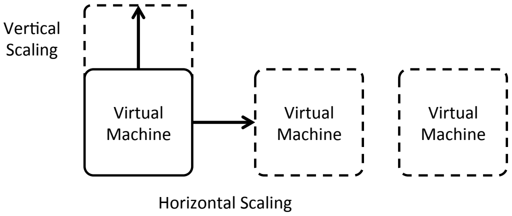

Fig. 3. Horizontal scaling and vertical scaling

In practice, however, only few cloud providers (e.g. ProfitBricks [49]) support vertical scaling whereas big providers such as Amazon Web Services[1] have not implemented vertical scaling yet [23]. Since private clouds and cloud providers such as Rackspace[2] and HP[3] apply the open-source platform OpenStack[4], it has become more and more prominent. Hence, this paper evaluates the feasibility of vertical scaling for OpenStack. For this purpose, this paper examines to what extent common guest operating systems, popular hypervisors, and OpenStack itself support vertical scaling.

The remainder of this paper is structured as follows. Section 2 presents related work before Section 3 assesses the support of vertical scaling by guest operating systems, hypervisors, and OpenStack. Finally, Section 4 presents a conclusion and discusses the fields of further research related to this work.

2 Related Work

We want to give an overview of OpenStack's vertical scaling capability. Yet, no literature exists that examines whether and to which extent OpenStack supports vertical scaling. However, several work has been done on horizontal and vertical scaling as well as combinations of both, often with respect to auto-scaling. Generally, authors use different approaches to scale a VM according to the current

[1] https://aws.amazon.com
[2] http://www.rackspace.com
[3] http://www.hpcloud.com
[4] https://www.openstack.org

demand. Lorido-Botrán et al. classify the available auto-scaling approaches in the following way: 1. static, threshold-based policies, 2. reinforcement Learning, 3. queuing theory, 4. control theory, and 5. time-series analysis [37].

For vertical scaling, one example is the work of Dawoud et al. who propose an architecture based on CPU utilization [19]. Similarly, Lu et al. present a tool for automatic vertical scaling of VMs based on throughput, average response time, and percentile response time [38]. Yazdanov and Fetzer apply reinforcement learning in their autonomic scaling framework [68]. Shen et al. propose a time-series-based system for vertical scaling of CPU and memory that considers VM migration and resource efficiency [57]. As presented by Rodero-Merino et al., vertical scaling is also possible through replacing an underlying server by a more powerful one [55]. Svärd et al. take vertical scaling even further by aggregating and pooling the resources that can be scaled vertically [61].

Other authors combine horizontal and vertical scaling. Caballer et al. suggest an architecture for horizontal and vertical scaling that adapts resources automatically to the requirements of scientific applications [13]. The framework from Dutta et al. uses a combination of horizontal and vertical scaling in order to optimize resource usage and incurring cost [22] while scaling decisions in the approach of Wang et al. are based on availability [66].

3 Vertical Scaling with OpenStack

In order to assess the support of vertical scaling in OpenStack, we examine the components (see Fig. 4) that are required to enable vertical scaling in OpenStack. Ideally, an administrator can use the dashboard of the cloud management platform in order to scale a running VM vertically. OpenStack as the *Cloud Management Platform* provides such a dashboard. The dashboard is connected to Open Stack's resource management that controls the *Hypervisor*. In the case of vertical scaling, the resource management that administers CPU, memory, disks, and network interfaces would prompt the hypervisor to provide a *Virtual Machine* more resources. The VM again would pass the resources to the *Guest Operating System (OS)* that runs within the VM. As a consequence, the applied guest OS, the used hypervisor, and OpenStack itself have to support vertical scaling, which is also known as hot plugging of resources, i.e. adding and removing of resources to or from a running VM [67]. Therefore, our assessment starts with the guest OSs (see Sec. 3.1) before we continue with the hypervisors (see Sec. 3.2). Finally, we examine OpenStack as cloud management platform that coordinates the scaling process with its dashboard (see Sec. 3.3).

Regarding the considered resource categories, we reviewed Amazon Web Services, Rackspace, Profitbricks, OpenStack[5] itself, and market places like the Deutsche Börse Cloud Exchange[6]. Knowing that other resource categories might be interesting for future work, we decide to consider the most important resources in this work and therefore focus our review on adding and removing computing

[5] https://www.openstack.org/software/openstack-compute/

[6] http://www.dbcloudexchange.com

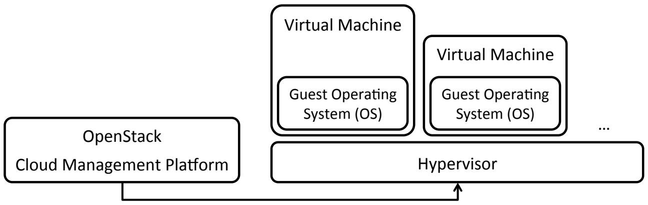
Virtual Machine
Guest Operating System (OS)
Virtual Machine
Guest Operating System (OS)
...
OpenStack
Cloud Management Platform
Hypervisor

Fig. 4. Overview of interaction between OpenStack, a hypervisor, virtual machines, and guest operating systems

power, memory space, and storage space as well as extending and shrinking the network. In this context, we distinguish between adding or removing an additional disk and extending or shrinking an existing disk because for the first case, the used hard drive might have to be changed for applications running inside the OS. Since the introduction of plug and play as well as of the universal serial bus (USB), adding and removing disks and network devices has become a common action. However, adding and removing disks and network devices is still important in this context because hypervisors have to support the addition and removal as well.

3.1 Vertical Scaling Capabilities of Guest Operating Systems

We use the most common operating systems, namely CentOS 6.5[7], SUSE Linux Enterprise Server 11 SP3[8], and Debian 7.4.0[9] as well as Ubuntu Server 14.04 LTS[10], and Microsoft Windows Server 2012 R2[11]. Our literature review is mainly based on websites, documentations, release notes, and blog entries in order to identify whether and to which extent an OS supports vertical scaling. Thereby, we classify the OS's support in four categories: yes, partly, no, and unknown. The results are shown in Table 1.

In general, all considered operating systems support vertical scaling with almost all considered types of resources. For CentOS, extending and shrinking is only possible for unmounted disks. SUSE Linux Enterprise Server does not offer the addition and removal of a CPU core as a full virtualization guest. Furthermore, a user of SUSE Linux Enterprise can extend and shrink disks with certain file systems. Similarly, Debian supports extending and shrinking a disk depending on its file system. Ubuntu Server supports vertical scaling of all resources except shrinking a disk, which works for unmounted file systems only. For Microsoft Windows Server, we cannot determine whether it supports the removal of a CPU core and memory. Furthermore, Microsoft Windows Server only supports

[7] https://www.centos.org

[8] https://www.suse.com/products/server/

[9] https://www.debian.org/index.en.html

[10] http://www.ubuntu.com/server

[11] http://www.microsoft.com/en-us/server-cloud/products/windows-server-2012-r2/

[12] [59] distinguishes SUSE running as paravirtualized guest and full virtualization guest. We focus on full virtualization here.

Table 1. Comparison of resources that can be vertically scaled with CentOS 6.5, SUSE Linux Enterprise Server 11 SP3, and Debian 7.4.0 as well as Ubuntu Server 14.04 LTS and Microsoft Windows Server 2012 R2

		CentOS	SUSE	Debian	Ubuntu	Windows Server
CPU core	Add/ Remove	Yes[52]/ Yes[52]	No[59]/ No[59][12]	Yes[3]/ Yes[3]	Yes[3]/ Yes[3]	Yes[64]/ ?
Memory	Add/ Remove	Yes[51]/ Yes[51]	Yes[59]/ Yes[59]	Yes[64]/ Yes[2]	Yes[64]/ Yes[2]	Yes[64]/ ?
Disk	Add/ Remove	Yes[50]/ Yes[50]	Yes[59]/ Yes[59]	Yes[6]/ Yes[20]	Yes[7]/ Yes[7]	Partly[45]/ Yes[45]
	Extend/ Shrink	Partly[53]/ Partly[53]	Partly[60]/ Partly[60]	Partly[5]/ Partly[5]	Yes[15]/ Partly[15]	Yes[43]/ Partly[43]
Network	Add/ Remove	Yes[50]/ Yes[50]	Yes[59]/ Yes[59]	Yes[9]/ Yes[9]	Yes[58]/ Yes[14]	Yes[44]/ Yes[44]

the addition of SCSI disks during run time and shrinking previously expanded disks.

The aforementioned review is based on certain assumptions. First, we focus on adding or removing hardware to or from a guest and not to or from the host. Second, if the addition or removal works for one architecture such as x86, our assumption is that it is also possible with other architectures. Third, we assume that an OS supports adding or removal of a certain hardware if we find that the OS is able to do so with a specific hypervisor. Furthermore, if an older version of an OS offer a certain functionality, we infer that newer versions offer the same functionality. Similarly, we consider the edition of an OS that offers the most functionalities, which often refers to the 64 bit version of an OS. Moreover, we do not require the considered OS to detect the changed hardware on its own but at most through a command in the command-line interface. Lastly, we make use of the fact that CentOS builds on the repository of Red Hat Enterprise Linux[13] by determining the functionality supported by CentOS through the functionality that Red Hat Enterprise Linux offers [17].

3.2 Vertical Scaling Capabilities of Hypervisors

As a next step, we review the most common hypervisors, namely KVM[14], VMware vSphere Hypervisor[15], Xen Project Hypervisor[16], and Microsoft Hyper-V[17]. Using the same four categories and similar sources as in Section 3.1,we determine whether and to which extent the hypervisors support vertical scaling. The results are shown in Table 2.

[13] http://www.redhat.com/products/enterprise-linux/

[14] http://www.linux-kvm.org

[15] http://www.vmware.com/products/vsphere-hypervisor

[16] http://www.xenproject.org/developers/teams/hypervisor.html

[17] http://technet.microsoft.com/en-us/library/hh831531.aspx

Table 2. Comparison of resources that can be vertically scaled with KVM, VMware vSphere Hypervisor, Xen Project Hypervisor, and Microsoft Hyper-V

		KVM	VMware	Xen	Hyper-V
CPU core	Add/ Remove	Yes[30]/ No[30]	Yes[65]/ No[65]	Yes[26]/ Yes[26]	No[34]/ No[34]
Memory	Add/ Remove	Yes[31,33]/ Yes[31,33]	Yes[65]/ No[65]	Yes[25]/ Yes[25]	Yes[42]/ No[42]
Disk	Add/ Remove	Partly[32]/ ?	Partly[4]/ Yes[4]	Yes[27]/ Yes[27]	Partly[54]/ Partly[54]
	Extend/ Shrink	No[56]/ No[56]	Yes[65]/ Partly[65]	Partly[40]/ No[40]	Partly[43]/ Partly[43]
Network	Add/ Remove	Yes[32]/ ?	Yes[65]/ Yes[63]	Yes[1]/ Yes[1]	No[34]/ No[54]

Generally, the considered hypervisors support vertical scaling with all considered resources. With KVM, removing a CPU core and altering disk size do not work. Furthermore, only SCSI disks can be added while we cannot determine the support for removing a disk and a network. Similarly, VMware does not support the removal of a CPU core and of memory. The addition of a disk also only works for SCSI disks. Additionally, while shrinking a disk, the VM stops responding. Xen enables vertical scaling with almost all resources. One exception is that Xen does not allow shrinking a disk and that extending a disk works only for certain file systems. For Hyper-V, the addition and removal of a CPU core and a network as well as the removal of memory are not available. Furthermore, only SCSI disks can be added and removed. Lastly, Hyper-V requires two functionalities in order to extend or shrink a virtual disk: Firstly, the disk has to be of a certain file format, i.e. the way how the disk's data and file format are stored. Secondly, the relevant disk has to be attached to a SCSI controller.

The aforementioned review is based on certain assumptions. When increasing or decreasing the memory assigned to a guest OS, KVM and Xen use memory ballooning, which takes memory from the guest OS and gives the memory to the host or vice versa [16]. Since memory ballooning is one method of vertical scaling, we consider it as equivalent to the addition and removal of memory. Lastly, we assume that tools that manage the interaction between the OS and the host (e.g. VMware Tools) are installed on the OS.

3.3 Vertical Scaling Capabilities of the Cloud Management Platform

Finally, we examine whether OpenStack supports vertical scaling. According to the OpenStack Operations Guide, OpenStack only supports horizontal scaling in order to match the cloud paradigm of "cloud-based applications that typically request more, discrete hardware". Consequently, the ideal for OpenStack is to add a VM and to load balance among the existing VMs in case of increasing resource demand [47].

However, OpenStack allows altering the size of an existing server during run time by changing the flavor [46]. A flavor is a hardware templates that defines a number of CPU cores, a memory size, a root disk size, and a data disk space. An administrative user can create, edit, and delete flavors. The default installation provides five different flavors [48]. Nevertheless, the question is whether changing the flavor is as quickly as hot plugging of resources. Therefore, we examine the source code of OpenStack. In the compute API file[18], we find that OpenStack resizes a VM by initiating a migration process. Shutting down the VM and moving it to a new host requires some time. Consequently, OpenStack does not provide vertical scaling within few seconds as provided by hot plugging and as presented in the introduction.

4 Conclusion and Future Work

This paper evaluates the vertical scaling capability of OpenStack. More specifically, this paper assesses whether the components that are relevant for enabling vertical scaling with OpenStack support vertical scaling. Based on an literature review, we evaluate common guest operating systems, popular hypervisors, and OpenStack regarding vertical scaling. Hereby, the paper gives a current overview of the resources supported for vertical scaling and allows for comparisons between the considered operating systems and between the hypervisors regarding vertical scaling.

In this paper, we identified that guest operating systems and hypervisors are in general capable of vertical scaling. OpenStack however does not support vertical scaling at all and is rather designed for horizontal scaling. Therefore, we recommend that future work focuses on how to implement vertical scaling in OpenStack. Possible solutions could be the removal of the flavor system or the automatic creation of flavors during run time as needed. In any case, the vertical scaling mechanism should be changed from migration to hot plugging. Additionally, we could not completely determine the vertical scaling capabilities of operating systems and hypervisors, which is thus left for future work. Furthermore, our work concentrated on the qualitative assessment of vertical scaling capabilities on a scale with yes, partial, and no. On these grounds, we suggest that future research performs a qualitative assessment of vertical scaling capabilities of operating systems, hypervisors, and OpenStack and defines metrics for the degree of support. Moreover, future research should examine how hypervisors, operating systems, and the implementation of vertical scaling in OpenStack influence the quality of vertical scaling in terms of spin-up time, speed, reliability etc. Lastly, vertical scaling is not always possible since the vertical scaling capacity is limited due to the limited resources of a VM. Therefore, our immediate future work will combine horizontal and vertical scaling in order to deal with demand changes more efficiently.

[18] https://github.com/openstack/nova/blob/master/nova/compute/api.py

Acknowledgment

This work is part of the project CLoUd Services Scalability (CLUSS) that is funded by the German Federal Ministry of Education and Research (funding code 01IS13013A-D). We also would like to thank Andreas Sperber from the gloveler GmbH for providing us with Figure 1.

References

1. Xen : Adding and removing NICs on virtual machines | linux - storage - virtualization, http://linux.cloudibee.com/2009/04/xen-adding-and-removing-nics-on-virtual-machines/
2. https://www.kernel.org/doc/Documentation/memory-hotplug.txt (2007), https://www.kernel.org/doc/Documentation/memory-hotplug.txt
3. Linux Hotplug a CPU (2009), http://www.cyberciti.biz/faq/debian-rhel-centos-redhat-suse-hotplug-cpu/
4. Vmware Linux Guest Add a New Hard Disk Without Rebooting Guest - nixCraft (2010), http://www.cyberciti.biz/tips/vmware-add-a-new-hard-disk-without-rebooting-guest.html
5. Chapter 12. Advanced Administration (2013), http://debian-handbook.info/browse/stable/advanced-administration.html
6. How to add a new hard disk or partition using UUID and ext4 filesystem (2014), http://www.debiantutorials.com/how-to-add-a-new-hard-disk-or-partition-using-uuid-and-ext4-filesystem/
7. QemuDiskHotplug - Ubuntu Wiki (2014), https://wiki.ubuntu.com/QemuDiskHotplug
8. Amazon Web Services: Auto Scaling, https://aws.amazon.com/autoscaling/
9. Aoki, O.: Debian GNU/Linux Reference - 10.8.2 Triggering network configuration – hotplug (2005), http://www.linuxtopia.org/online_books/linux_system_administration/debian_linux_guides/debian_linux_reference_guide/ch-gateway.en_024.html
10. Armbrust, M., Fox, A., Griffith, R., Joseph, A.D., Katz, R., Konwinski, A., Lee, G., Patterson, D., Rabkin, A., Stoica, I., Zaharia, M.: A View of Cloud Computing. Communications of the ACM 53(4), 50–58 (2010)
11. Baun, C., Kunze, M., Nimis, J., Tai, S.: Cloud Computing: Web-Based Dynamic IT Services. Springer Berlin Heidelberg (2011)
12. Bodík, P., Griffith, R., Sutton, C., Fox, A., Jordan, M., Patterson, D.: Statistical machine learning makes automatic control practical for internet datacenters. In: Proceedings of the 2009 conference on Hot topics in cloud computing (2009)
13. Caballer, M., García, A., Moltó, G., de Alfonso, C.: Towards SLA-driven Management of Cloud Infrastructures to Elastically Execute Scientific Applications. In: 6th Iberian Grid Infrastructure Conference (IberGrid). pp. 207–218 (2012)
14. Canonical: Ubuntu Manpage: ifconfig - configure a network interface, http://manpages.ubuntu.com/manpages/precise/man8/ifconfig.8.html
15. Canonical: Ubuntu Manpage: resize2fs - ext2/ext3/ext4 file system resizer (2010), http://manpages.ubuntu.com/manpages/precise/en/man8/resize2fs.8.html
16. Capitulino, L.: Automatic ballooning. Tech. rep. (2013), http://www.linux-kvm.org/wiki/images/f/f6/Automatic-ballooning-slides.pdf
17. CentOS: About - CentOS Wiki (2012), http://wiki.centos.org/About

18. Chieu, T.C., Mohindra, A., Karve, A.A., Segal, A.: Dynamic Scaling of Web Applications in a Virtualized Cloud Computing Environment. In: 2009 IEEE International Conference on e-Business Engineering. pp. 281–286 (2009)
19. Dawoud, W., Takouna, I., Meinel, C.: Elastic virtual machine for fine-grained cloud resource provisioning. In: Krishna, P.V., Babu, M.R., Ariwa, E. (eds.) Global Trends in Computing and Communication Systems, pp. 11–25. Springer Berlin Heidelberg (2012)
20. Debian Help: Mount Umount Harddisk Floppy CDROM under Debian, http://www.debianhelp.co.uk/mount.htm
21. Douglas, P.: How Stephen Fry takes down entire websites with a single tweet | News | TechRadar (2010), http://www.techradar.com/news/internet/how-stephen-fry-takes-down-entire-websites-with-a-single-tweet-674170
22. Dutta, S., Gera, S., Verma, A., Viswanathan, B.: SmartScale: Automatic Application Scaling in Enterprise Clouds. In: 2012 IEEE Fifth International Conference on Cloud Computing. pp. 221–228 (2012)
23. Galante, G., De Bona, L.C.E.D., Mury, A.R., Schulze, B.: Are Public Clouds Elastic Enough for Scientific Computing? In: 3rd International Workshop on Cloud Computing and Scientific Applications (CCSA) (2013)
24. Google: Web Apps Articles & Tutorials — Google Cloud Platform, https://cloud.google.com/developers/articles/auto-scaling-on-the-google-cloud-platform
25. Herron, K.: How to Hot Add/Remove Memory from a Xen Domain | Backdrift, http://backdrift.org/xen-memory-hot-add-and-remove
26. Herron, K.: How to Hot Add/Remove VCPUs from a Xen Domain | Backdrift, http://backdrift.org/how-to-hot-addremove-vcpus-from-a-xen-domain
27. Herron, K.: Xen Disk Hot Add (Block Device) Howto | Backdrift, http://backdrift.org/xen-disk-hot-add-block-device-howto
28. Islam, S., Keung, J., Lee, K., Liu, A.: Empirical prediction models for adaptive resource provisioning in the cloud. Future Generation Computer Systems 28(1), 155–162 (2012)
29. Klems, M.: Experiment-Driven Quality Assessment and Optimization of Cloud-Based Distributed Database Systems. Software Quality Professional Magazine 15(4) (2013)
30. KVM: CPUHotPlug - KVM, http://www.linux-kvm.org/page/CPUHotPlug
31. KVM: FAQ - KVM, http://www.linux-kvm.org/page/FAQ
32. KVM: Hotadd pci devices - KVM, http://www.linux-kvm.org/page/Hotadd_pci_devices
33. KVM: NetworkingTodo - KVM, http://www.linux-kvm.org/page/NetworkingTodo
34. Laverick, M.: Hyper-V R2eality: VMs not so hot after all. . . « Mike Laverick. . . , http://www.mikelaverick.com/2014/02/hyper-v-r2eality-vms-not-so-hot-after-all/
35. Lenk, A., Pallas, F.: Modeling Quality Attributes of Cloud-Standby-Systems. In: Lau, K.K., Lamersdorf, W., Pimentel, E. (eds.) Service-Oriented and Cloud Computing, pp. 49–63. Springer Berlin Heidelberg (2013)
36. Lenk, A., Tai, S.: Cloud Standby: Disaster Recovery of Distributed Systems in the Cloud. In: Villari, M., Zimmermann, W., Lau, K.K. (eds.) Service-Oriented and Cloud Computing, pp. 32–46. Springer Berlin Heidelberg (2014)
37. Lorido-Botrán, T., Miguel-Alonso, J., Lozano, J.A.: Auto-scaling techniques for elastic applications in cloud environments. Tech. rep., Department of

Computer Architecture and Technology University of the Basque Country (2012), http://www.sc.ehu.es/ccwbayes/isg/administrator/components/com_jresearch/files/publications/autoscaling.pdf

38. Lu, L., Zhu, X., Griffith, R., Padala, P., Parikh, A., Shah, P., Smirni, E.: Application-Driven Dynamic Vertical Scaling of Virtual Machines in Resource Pools. In: IEEE/IFIP Network Operations and Management Symposium (NOMS 2014) (2014)
39. Mao, M., Humphrey, M.: A Performance Study on the VM Startup Time in the Cloud. In: 2012 IEEE Fifth International Conference on Cloud Computing. pp. 423–430 (2012)
40. McWilliams, G.: Grant McWilliams - Resize Disk Image used as Xen DomU partition (2009), http://grantmcwilliams.com/item/265-resize-disk-image-used-as-xen-domu-partition
41. Mell, P., Grance, T.: The NIST Definition of Cloud Computing. Special Publication 800-145, National Institute of Standards and Technology (NIST) (2011), http://csrc.nist.gov/publications/nistpubs/800-145/SP800-145.pdf
42. Microsoft: Hyper-V Dynamic Memory Overview, http://technet.microsoft.com/en-us/library/hh831766.aspx
43. Microsoft: Online Virtual Hard Disk Resizing Overview, http://technet.microsoft.com/en-us/library/dn282286.aspx
44. Microsoft: Networking (2008), http://technet.microsoft.com/en-us/library/cc772351(v=ws.10).aspx
45. Microsoft: Manage Server Storage in Windows Server Essentials (2014), http://technet.microsoft.com/en-us/library/dn550731.aspx
46. OpenStack Foundation: Change the size of your server - OpenStack End User Guide - current, http://docs.openstack.org/user-guide/content/nova_cli_resize.html
47. OpenStack Foundation: Chapter 5. Scaling - OpenStack Operations Guide, http://docs.openstack.org/trunk/openstack-ops/content/scaling.html
48. OpenStack Foundation: Manage flavors - OpenStack Admin User Guide - current, http://docs.openstack.org/user-guide-admin/content/dashboard_manage_flavors.html
49. ProfitBricks: Live Vertical Scaling. Tech. rep. (2014), https://www.profitbricks.co.uk/sites/default/files/documents/LiveVerticalScalingbyProfitBricks.pdf
50. Red Hat: Red Hat Enterprise Linux 6 6.0 Release Notes. Tech. rep. (2010), https://access.redhat.com/site/documentation/en-US/Red_Hat_Enterprise_Linux/6/pdf/6.0_Release_Notes/Red_Hat_Enterprise_Linux-6-6.0_Release_Notes-en-US.pdf
51. Red Hat: Red Hat Enterprise Linux 5 5.0 Release Notes. Tech. rep. (2012), https://access.redhat.com/site/documentation/en-US/Red_Hat_Enterprise_Linux/5/pdf/5.0_Release_Notes/Red_Hat_Enterprise_Linux-5-5.0_Release_Notes-en-US.pdf
52. Red Hat: Red Hat Enterprise Linux 6 6.5 Release Notes. Tech. rep. (2013), https://access.redhat.com/site/documentation/en-US/Red_Hat_Enterprise_Linux/6/pdf/6.5_Release_Notes/Red_Hat_Enterprise_Linux-6-6.5_Release_Notes-en-US.pdf
53. Red Hat: Red Hat Enterprise Linux 6 Storage Administration Guide. Tech. rep. (2013), https://access.redhat.com/documentation/en-US/Red_Hat_Enterprise_Linux/6/pdf/Storage_Administration_Guide/Red_Hat_Enterprise_Linux-6-Storage_Administration_Guide-en-US.pdf

54. Remde, K.: Kevin Remde's Full of I.T., where we discuss Windows Server, System Center, Cloud, Virtualization, and all things Microsoft for the IT Professional and IT Manager., http://blogs.technet.com/b/kevinremde/archive/2013/03/05/you-want-to-hot-add-what-20-days-of-server-virtualization-part-2-of-20.aspx
55. Rodero-Merino, L., Vaquero, L.M., Gil, V., Galán, F., Fontán, J., Montero, R.S., Llorente, I.M.: From infrastructure delivery to service management in clouds. Future Generation Computer Systems 26(8), 1226–1240 (2010)
56. Server Fault: linux - KVM online disk resize? - Server Fault, http://serverfault.com/questions/122042/kvm-online-disk-resize
57. Shen, Z., Subbiah, S., Gu, X., Wilkes, J.: Cloudscale: elastic resource scaling for multi-tenant cloud systems. In: Proceedings of the 2nd ACM Symposium on Cloud Computing (SOCC '11) (2011)
58. Slonka, K.J.: /public/slonkak (2013), http://www.kevinslonka.com/index.php?section=1&blog=305
59. SUSE: SUSE Linux Enterprise Server: Supported Virtualization Technologies. Tech. rep. (2013), http://www.novell.com/docrep/2009/09/SUSE_Linux_Enterprise_Server_Virtualization_Technology_Support_en.pdf
60. SUSE: Technical Information | SUSE Linux Enterprise Server (2014), https://www.suse.com/products/server/technical-information/
61. Svärd, P., Hudzia, B., Tordsson, J., Elmroth, E.: Hecatonchire: Enabling Multi-Host Virtual Machines by Resource Aggregation and Pooling. Tech. rep., UmeåUniversity (2014), http://www8.cs.umu.se/research/uminf/reports/2014/011/part1.pdf
62. Urgaonkar, B., Pacifici, G., Shenoy, P., Spreitzer, M., Tantawi, A.: An analytical model for multi-tier internet services and its applications. In: Proceedings of the 2005 ACM SIGMETRICS International Conference on Measurement and Modeling of Computer Systems. pp. 291–302 (2005)
63. VMware: vSphere Networking vSphere 5.5. Tech. rep. (2013), http://pubs.vmware.com/vsphere-55/topic/com.vmware.ICbase/PDF/vsphere-esxi-vcenter-server-55-networking-guide.pdf
64. VMware: Guest OS Compatibility Guide. Tech. rep. (2014), http://partnerweb.vmware.com/comp_guide2/pdf/VMware_GOS_Compatibility_Guide.pdf
65. VMware: vSphere Virtual Machine Administration ESXi 5.5. Tech. rep. (2014), http://pubs.vmware.com/vsphere-55/topic/com.vmware.ICbase/PDF/vsphere-esxi-vcenter-server-551-virtual-machine-admin-guide.pdf
66. Wang, W., Chen, H., Chen, X.: An Availability-Aware Virtual Machine Placement Approach for Dynamic Scaling of Cloud Applications. In: 2012 9th International Conference on Ubiquitous Intelligence and Computing and 9th International Conference on Autonomic and Trusted Computing. pp. 509–516 (2012)
67. Yazdanov, L., Fetzer, C.: Vertical Scaling for Prioritized VMs Provisioning. In: 2012 Second International Conference on Cloud and Green Computing. pp. 118–125 (2012)
68. Yazdanov, L., Fetzer, C.: VScaler: Autonomic Virtual Machine Scaling. In: 2013 IEEE Sixth International Conference on Cloud Computing. pp. 212–219 (2013)