The ASCETiC Testbed - An Energy Efficient Cloud Computing Environment

Marc Körner(✉), Alexander Stanik, Odej Kao, Marcel Wallschläger, and Sören Becker

Department of Telecommunication Systems, Complex and Distributed IT Systems, Technische Universität Berlin, Einsteinufer 17, 10587 Berlin, Germany
marc.koerner@tu-berlin.de

Abstract. Nowadays, the energy consumption of data centers is one of the biggest challenges in order to reduce operational expenditure (OPEX) and the carbon dioxide footprint. Most efforts are investigating the modernization of air conditions and server hardware, but also the optimization of resource allocations. Moreover, virtual server are migrated from one physical host to another in order to be able to shutdown unused physical computer nodes or even an entire rack. The Adapting Service lifeCycle towards EfficienT Clouds (ASCETiC) project tries to tackle this issue in another way. This project develops a toolbox that provides libraries and components which can be used to develop energy efficient cloud software on all three layers of the usual could stack. Thus, power consumption can be reduced by deploying energy efficient software in the cloud. This paper presents the innovative ASCETiC testbed located at the Technical University in Berlin (TUB), which is the deveopment and evaluation eviroment for the afore metioned software.

Keywords: Testbed · Cloud computing · Energy-aware · Software-defined networking

1 Introduction

In the recent years the information technology (IT) related power consumption world wide is growing rapidly. One of the biggest impact factors are data centers. They provide several IT services to customers and end-users. With the upcoming of cloud computing many companies strive to outsource their services into the cloud. However, there is an acceleration of adoption of cloud applications and services by enterprises [9]. Consequently, experts warn of a dramatic increase in energy consumption for cloud computing. Although cloud computing offers the potential for energy saving through the centralization of computing and storage technologies in large data centers, a few mechanisms to reduce energy consumption have been already exploited so far.

The ASCETiC project [5] is focused on providing novel Cloud Computing (CC) methods and tools in order to support software developers optimizing

S. Guo et al. (Eds.): TridentCom 2016, LNICST 177, pp. 93–102, 2017.
DOI: 10.1007/978-3-319-49580-4_9

energy efficiency within their applications. This minimizes the carbon dioxide footprint resulting from designing, developing, deploying and running software in Clouds, while maintaining other quality aspects of software to adequate and agreed levels. In particular, the main objectives of the ASCETiC project are:

(i) to develop models for green and efficient software design, supporting sustainability and high quality of service levels at all stages of software development and execution
(ii) to develop and to evaluate a framework with identified energy efficiency parameters and metrics for cloud services
(iii) to develop methods for measuring, analyzing, and evaluating energy use in software development and execution, complementing quality measures
(iv) to integrate energy efficiency at the same level as other quality aspects during service requirement, design, construction, deployment, and operation leading to an Energy Efficiency Embedded Service Lifecycle

Therefore, ASCETiC is delivering a framework composed out of tools and methods covering each phase of development lifecycle for implementing energy efficient cloud services at all layers of the cloud stack [2]. Thus, the applicability of ASCETIC to achieve 20 % of energy savings will be demonstrated on two industry use cases. In order to reach this goal, ASCETiC initiative measures how software systems actually use cloud resources, in order to optimize consumption the resource consumption. Moreover, the awareness of the amount of energy needed by software will help in learning how to target software optimization where it provides the greatest energy returns. Therefore, all three layers of the cloud computing stack: Software-as-a-Service (SaaS), Platform-as-a-Service (PaaS) and Infrastructure-as-a-Service (IaaS) have to implement a MAPE (Monitor, Analyze, Plan and Execute) loop. Each layer monitors relevant energy efficiency status information locally and shares this with the other layers, assesses its current energy status and forecasts future energy consumption as needed.

This paper basically presents the testbed which is used in the ASCETiC project. This testbed provides an advanced OpenStack based cloud with additional energy monitoring capabilities. In particular, it presents the MAPE (Monitoring, Analysis, Planning, Execution) on the IaaS layer.

The remaining paper is structured as follows. Section 2 provides a brief background analysis which is followed by an introduction of the ASCETiC software stack, in Sect. 3. Afterwards the testbed architecture is introduced followed by the conclusion and futuer work, in Sect. 5. Finally a disclaimer and acknowledgment is given.

2 Background

The CC paradigm itself was generally evolving the scalability of resource allocation by virtualization. Moreover, CC is an opportunity to massively reduce OPEX costs and save energy. Overall, cloud computing provides the opportunity to save 70–90% direct energy for servers and cooling while increasing the network energy consumption by only 2–3% [7]. This demonstrates that a total energy saving of 68–87% is feasible by just switching to CC. Beside energy savings due to the CC paradigm another contemporary research challenge [1] is to use this model an integrate energy saving mechanism within and amongst the cloud layers [6]. This has the potential to increase the entire impact of this particular paradigm. To support this kind of research specific testbeds are needed. They need to provide particular hardware informations to the different components working within the cloud stack. Furthermore, an API and specific hardware functions are required to support the cloud stack based energy adaptation mechanisms.

The current State of the Art (SotA) process regarding compute and networking hardware is highly connected to Moore's Law [8]. New hardware becomes more and more energy efficient form year to year. The criteria with the almost biggest impact on the awareness of these new energy efficient demands is the hardware integration density. This process is amongst others driven by mobile devices and the approach to improve their runtime. However, this effects also regular compute hardware driven by the increasing amount of data centers and the cost-value ratio.

The ASCETiC project consortium has identified some weak points concerning the used cluster during the first project year evaluation. Unfortunately the deployed hardware was not contemporary and unable to supporting new energy saving modes. Moreover, recent hardware is also supporting specific virtualization need and provide a better scalability in this area. To overcome this issue, the first year's testbed is migrated and redeployed on SotA cluster as described in the following sections. Lessens learned during the first deployment cycles were taken into account in combination with the new identified project requirements [3] in order to design the new testbed. In particular, the aim is to support a self-adaptable optimization approach for clouds. This includs trade-offs on energy consumption, maintainability, and supportability, which builds the umbrella for the improved design.

3 The ASCETiC Toolbox

The ASCETiC Toolbox is the core component developed within the project. It is utilizing the ASCETiC testbed in order to Provide an energy efficient CC stack by using the innovative cloud testbed capabilities. It delivers the necessary mechanisms for developers and engineers to develop energy efficient applications and modules on all layers of the cloud stack. Furthermore, the testbed itself is hosting the ASCETiC toolbox, depict in Fig. 1. This is a collection of modules which are using the innovative testbed capabilities to implement the SaaS, PaaS, and IaaS stack. This includes energy awareness and self-adaptation mechanisms on all of the afore mentioned layers. The ASCETiC toolbox and its source code is available on the project website [5] and on github.

Section 4 will further introduce the ASCETiC testbed as part of this toolbox. In particular, the infrastructure manager (IM) and the distributed file system

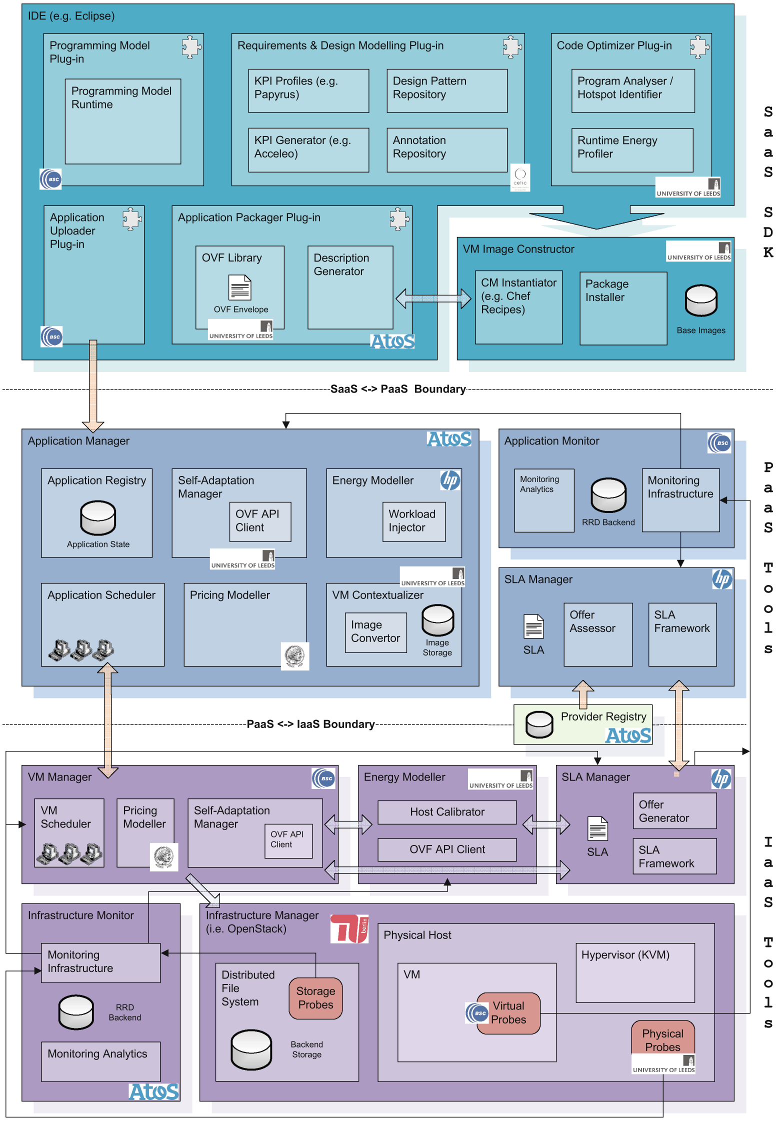

Fig. 1. ASCETiC software toolbox

(DFS). Moreover, some specific components which to obtain physical and virtual energy probes on the IaaS layer will be introduced. Thus, the next section will present the hardware and software architecture used for the ASCETiC testbed deployment at TUB.

4 Testbed Architecture

The main objective for the testbed architecture is to revise the physical infrastructure used in the first year and taking into account lessons learned during the first operational phase [4]. Moreover, the planned testbed improvements were designed to deliver an appropriate support for the ASCETiC business use-cases and further improve the sustainability of the Infrastructure Manager (IM) and the underlying cluster. In addition, the revised cluster architecture is also supposed to support full and fine-grained control over the networking resources, which further offers the opportunity to evaluate energy influences in this area. Therefore the entire cluster deployment is revised, improved and extended with:

(i) more contemporary compute hardware - from the old 16 node based Asok (10 years old cluster) to the 200 nodes Wally cluster, which will be further introduced in this section
(ii) increase the total amount of nodes used in the IM and the resulting experimental scalability
(iii) integrate Software Defined Networking (SDN) resources for a comprehensive cluster network approach
(iv) a second IM instance in order to have two environments (a small testing and large stable)

The IM is generally responsible for the entire cluster management. It is used to provide an API for the ASCETiC toolbox virtual machine manager (VMM) to allocate, parameterize and place virtual machines (VM) within the cluster infrastructure. The novelty beyond the state of the art (StoA) is a comprehensive consideration of all its cloud managed resources (compute, storage, and network) in combination with dynamic energy consumption measurements and runtime probes in order to provide a foundation for determining an energy aware cloud model. The IaaS provides power consumption probes for VMs, physical nodes, and host groups. This approach requires access to a fine-grained resource control in order compute these informations and save energy. In other words, the IM controls all resources provided by the revised and enhanced cluster testbed. It is the central management entity for all within the cluster available resources. It also manages all VM operations, from the deployment over the migration up to un-deployment. It's API is used as a communication interface for the VMM, which delegates the management interaction and uses the IM execution unit.

The utilized wally cluster is composed out of 200 heterogeneous nodes and one master node. The nodes are numbered from wally001 till wally200 and have a FQDN according to their number: <NodeName>.cit.tu-berlin.de. They are interconnected on the first Ethernet interface by one large HP5400 switch, which also provides Internet connectivity. 50 nodes on this cluster are permanently allocated in order to create the ASCETiC energy efficient cloud testbed and support the project with and appropriated test and evaluation environment.

The testbed installation is separated into two different groups or more precisely two OpenStack instances, as depict in Fig. 2. The first "unstable" instance

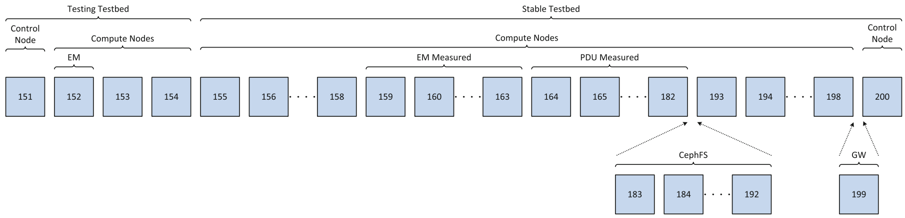

Fig. 2. Testbed environment

is for software tests with regards to the ASCETiC toolbox, while the second stable one is used to collected reference values with the three ASCETiC business use cases. One of the use cases is even deploying a large number of VMs in order to process thousands of jobs on the testbed. This is actually one of the important parameters indicating if the ASCETiC toolbox's adaption mechanisms are working. However, both OpenStack instances are using the same distribute file system also located within the allocated cluster partition. The physical deployment is composed out of the hardware listed in Table 1. In order to cover the SotA requirements, the wally cluster part belonging to the ASCETiC testbed was additionally equipped with SDN enabled network devices, as listed in Table 2.

Table 1. Wally cluster node hardware

Component	Model
CPU	Quadcore Intel Xeon CPU E3-1230 V2 3.30 GHz
Memory	16 GB RAM
Hard disk	3 × 1 TB
Network	2 × 1 GBit Ethernet NIC

Table 2. Wally cluster network hardware

Component	Model	Comment
TUB Mgmt network	HP 5400 legacy switch	public IP space with TUB addresses for the host node communication
VM Data network	2x NEC IP8800	SDN based OpenStack VM network using private IP addresses, for the VM to VM communication

The network related software deployment is composed out of the OpenStack Ice House distribution including Neutron and the OpenDaylight controller. This extends the previous approach and deliver the foundation for an additional energy aware network resource management. The VMs can be externally accessed via Network Address Translation (NAT) based user defined floating IPs. The physical hosts again can be accessed for administrative purposes over the regular TUB internal network on eth0 or the SDN data network using a private class C network addresses on eth1.

In the following subsections all testbed components will be introduced:

Testbed gateway and control node. The wally199 node is used as testbed access point. It provides all services which are not directly related to OpenStack. It hosts a layer three OpenVPN server. This VPN server is used by the clients to create their tunnel connection. It presents the central entry point for all external connections. Moreover, all traffic of the three different networks is basically routed there. TUB is also running a Chef server on this node which allows to deploy operating system (OS) images on the physical hosts on demand. This is required to e.g. roll out updates or testbed related modifications. Moreover, the on this machine installed Zabbix server is monitoring all testbed parameters including power probes from the physical and virtual machines. These values are further stored in a MySQL data-base and can be queried by modules inside the IaaS ASCETiC Toolbox layer.

Stable testbed. The stable testbed is used to deploy the use cases and explore the energy saving mechanisms implemented within the ASCETiC toolbox. The wally200 node is the cloud controller for the stable testbed. It hosts all management related OpenStack components like Neutron, Glance, and Keystone. This controller is managing a total amount of 32 compute nodes (Table 3).

Table 3. Stable testbed node distribution

Node	Comment
wally200	control node
wally193 - wally198	currently unmeasured compute nodes
wally157 - wally163	energy-meter measured compute nodes
wally164 - wally182	PDU measured compute nodes (in work)

All energy probes or more precisely power measurements are directly collected with customized scripts and pushed to zabbix, where they are available for the ASCETiC IaaS components like e.g. the VMM based VM scheduling and placement.

Testing testbed. The testing testbed serves as playground for the ASCETiC toolbox developers. It is an distinguish development environment where new implemented features can be tested without influencing the execution of the use cases running on the stable testbed. It is composed out of one master and three compute modes, as listed in Table 4.

CephFS. As distribute file system (DFS) and basic enabler for VM live migrations the testbed uses the latest from source build CephFS. The deployed CephFS installation on the testbed is composed out of ten nodes (wally183 - wally192). It uses five monitor daemons and and two one Terra-byte disks participating in

Table 4. Testing testbed node distribution

Node	Comment
wally155	control node
wally152	energy-meter measured compute node
wally153, wally154	currently unmeasured compute nodes

the DFS per node with round about 14 Terra-byte total DFS volume. In addition, every DFS node is further equipped with an 512 MB SSD, which is used as CephFS journals to increase the IOPS performance.

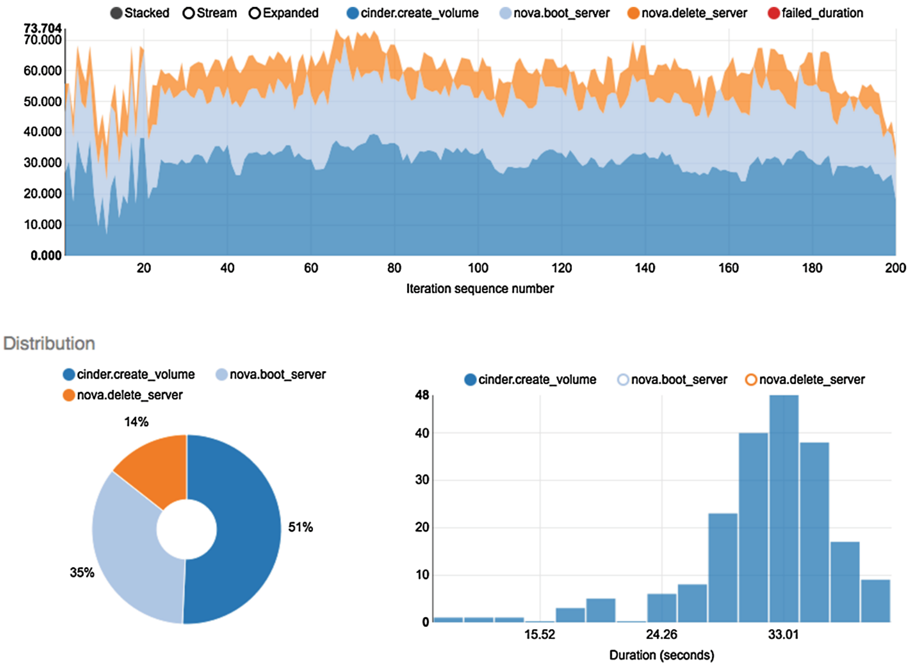

Fig. 3. Rally evaluation with 200 iterations using CephFS

A brief evaluation of the stable testbed instance with Rally [10] indicates that the deployment at TUB is working flawless. Spawning VMs with a typical Ubuntu 14.04 LTS image demonstrates that the usage of CephFS is a reasonable approach for IaaS providers to increase performance. The installation was benchmarked with a VM boot and delete process with 200 iterations running 20 concurrent deployments. As shown in Fig. 3, the testbed needs around 33 s to boot the VM. In contrast, booting from the local hard disk takes around 200 s, as depict in Fig. 4.

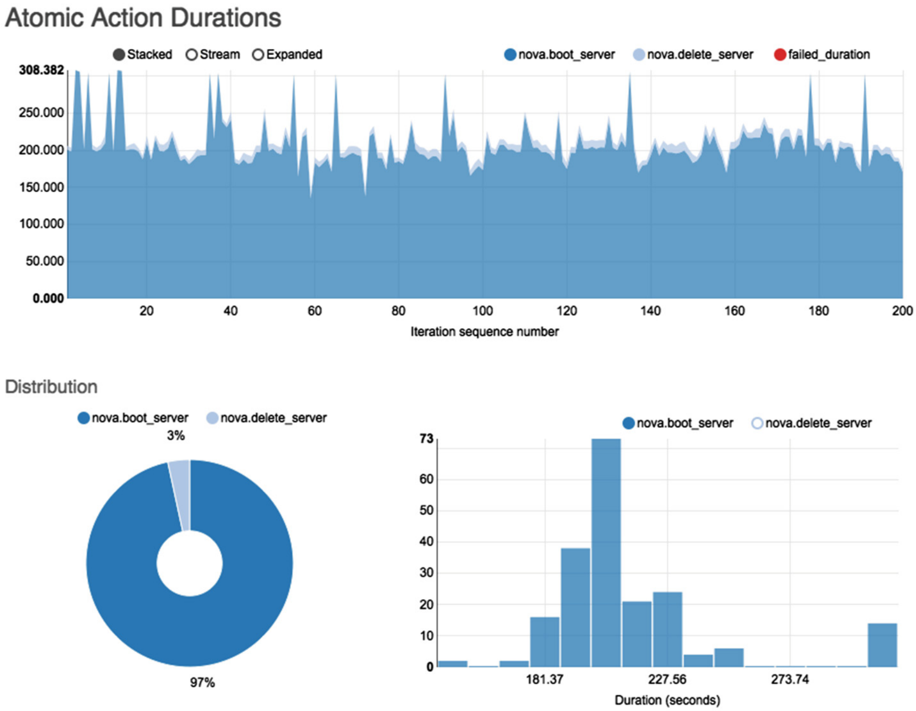

Fig. 4. Rally evaluation with 200 iterations using the nodes local hard disk

5 Conclusion and Future Work

The currently deployed testbed presents an innovative environment for researchers to evaluate energy efficient cloud computing on all cloud layers. The ASCETiC project is currently in its third and last project year. The testbed meanwhile supports a lot of energy saving features provided by the second year ASCETiC toolbox. Nevertheless, some ambitions testbed related improvements and support opportunities are currently in progress. In particular, the TUB team is currently working on an IPMI driver to control hosts automatically. This driver will basically provide the opportunity to dynamically turn physical hosts on and off on demand. This will improve the overall energy consumption. Moreover, the TUB cluster is using smart power distribution units (PDUs) which provide a similar opportunity. Thus, with a PDU driver it would be possible to compare host and host group energy values. Furthermore, the integration of an OpenFlow capable network monitoring tool enables the opportunity to consider network streams and VM distances for the VMM related scheduling, deployment, and migration process.

6 Disclaimer and Acknowledgments

The research leading to the presented results was partially funded by the European Commission's Seventh Framework Programme (FP7) under the Information and Communications Technologies (ICT) - Internet of Services, Software and Virtualisation under the grant agreement number 610874. The views and conclusions contained herein are those of the authors and should not be interpreted as necessarily representing the official policies or endorsements, either expressed or implied, of the ASCETiC project or the European Commission.

Minor paragraphs in the introduction of this paper are slightly modified citations from [3]. Thus, we would like to thank the entire ASCETiC Project Consortium for the original paperwork.

References

1. Buyya, R., Beloglazov, A., Abawajy, J.H.: Energy-efficient management of data center resources for cloud computing: A vision, architectural elements, and open challenges. CoRR, abs/1006.0308 (2010)
2. ASCETiC Project Consortium. Ascetic whitepaper, September 2014. https://www.cetic.be/IMG/pdf/whitepaper_2014_v0.9.9.pdf
3. ASCETiC Project Consortium. D2.2.2 architecture specification - version 2, December 2014. http://www.ascetic-project.eu/sites/default/ascetic/files/content-files/articles/D2.2.2ASCETiCArchitectureSpecification.pdf
4. ASCETiC Project Consortium. D6.1.1 testbed status report - version 1, October 2014. http://www.ascetic-project.eu/sites/default/ascetic/files/content-files/articles/ASCETiCD6.1.1Testbedstatusreportv1.0.pdf
5. ASCETiC Project Consortium. Ascetic, August 2015. http://ascetic.eu
6. Djemame, K., Armstrong, D., Kavanagh, R.E., Ferrer, A.J., Perez, D.G., Antona, D.R., Deprez, J.-C., Christophe Ponsard, D., Ortiz, M.M., Guitart, J., Lordan, F., Ejarque, J., Sirvent, R., Badia, R.M., Kammer, M., Kao, O., Agiatzidou, E., Dimakis, A., Courcoubetis, C., Blasi, L.: Energy efficiency embedded service lifecycle: towards an energy efficient cloud computing architecture. In: Joint Workshop Proceedings of the 2nd International Conference on ICT for Sustainability 2014, Stockholm, Sweden, 24–27 August 2014, pp. 1–6 (2014)
7. Google. Google apps: Energy efficiency in the cloud, September 2015. http://www.google.com/green/pdf/google-apps.pdf
8. Keyes, R.W.: The impact of Moore's Law. Solid-state Circ. Soc. Newsl. **11**(5), 25–27 (2006)
9. Pettey, C., van der Meulen, R.: Gartner says public cloud services are simultaneously cannibalizing and stimulating demand for external it services spending. Gartner Inc., Stamford, Press release, November 2012
10. Rally Project Core Team.: Rally, January 2016. https://wiki.openstack.org/wiki/Rally