

Implementing Educlouds using virtualization with Cloud Mashup's

D. Madhuri Devi

Student, Sri Sai Aditya Institute of Science and Technology, Surampalem, East Godavari (Dt.)

G. Santhi

Asst. professor, Sri Sai Aditya Institute of Science and Technology, Surampalem, East Godavari (Dt.)

ABSTRACT

In recent years, technologies have been introduced offering a large amount of computing and networking resources. Traditional application integration technology takes a long time to build and deploy, requiring professional developers and domain experts. They are server-centric and thus do not fully utilize the computing power and storage capability of client systems. Cloud computing is a new infrastructure deployment environment that delivers on the promise of supporting on-demand services like computation, software and data access, storage etc. Clouds provide inexpensive access to remote resources. Academia has not remained unaware of this trend, and several educational solutions (LMS) based on cloud technologies are already in place, especially for software as a service (SAAS) cloud. Extending the functionality to infrastructure and platform clouds (IAAS,PAAS) has not been explored yet. Learning process in academia has different stages and there is no design to classify clouds for each of them especially for computer science students. In this paper, we use the architecture and the organization of a Mashup Container that supports the deployment and the execution of Event Driven Mashups. In collaboration with PaaS, Virtualization provides an opportunity for extension of independent virtual resources based on available physical systems. Finally, the results of virtualization of mashup container through its supporting scalability and fault tolerance in cloud computing environment.

Keywords

Cloud Mashup, Virtualization.

1. INTRODUCTION

The face of the Internet is continually changing, as new services and novel applications appear and become globally noteworthy at an increasing pace. Computing and communication are being blended into new ways of using networked computing systems. The network retains its traditional role as a means of information exchange but, at the same time, it is also perceived as a means of sharing resources, where communication networks are used to share programs, information, processing capacity, communication, storage, and media. Next generation network and service infrastructures should overcome the scalability, flexibility, resilience, mobility and security bottlenecks of current network and service architectures, in order to provide a large variety of business models capable of dynamic and seamless end-to-end utilization of shared resources across a multiplicity of devices, networks, providers and service domains[1].

The concept of virtualization constitutes a promising one for addressing the requirements for the future of the Internet. With virtualization, multiple virtual networks are established over a shared, physical infrastructure, and each virtual network appears to have its own unique set of properties and performance characteristics. Virtualization exists in the context of computing as well as in networks, but current

virtualization techniques suffer from being based on managed configurations with static properties.

Virtualization is mainly used for the sharing of resources in restricted user communities such as Grid networks and server farms, while in networking, it is primarily used for virtual private network services. The Service Oriented Architecture (SOA) can be considered as a philosophy or paradigm to organize and utilize services and capabilities that may be under the control of different ownership domains.

In this paper, we present the Cloud Computing Architecture with its Service Models which formally introduces Virtualization technique and its system architecture. Later in detail we discuss about Mashup container, its requirements and management in a cloud computing environment. The Mashup Container is supposed to provide some mechanisms to support the management of Mashups usage, fault tolerance, scalability and Security is probably the most important aspect.

2. RELATED WORK

A) Service-Oriented Architecture

The SOA concept has been receiving considerable attention in recent years. According to the Organization for the Advancement of Structured Information Standards (OASIS), SOA can be defined as “a paradigm for organizing and utilizing distributed capabilities that may be under the control of different ownership domains. It provides a uniform means to offer, discover, interact with, and use capabilities to produce desired effects consistent with measurable preconditions and expectations”. Services can be described and accessed using well defined service interfaces and via standard message-exchanging protocols and exercised by following a service contract with certain policies. This provides a loose coupling of services and ensures operational agility.

B) Network resource virtualization and control

Virtualization constitutes the enabling technology for interconnecting groups of network nodes into virtual private networks (VPNs) [11]. According to [10], in a virtualized network, virtual networks exist co- instantaneously on top of a shared substrate. Different virtual networks may use different protocols and packet formats and provide alternate packet delivery systems. For example over a MPLS/GMPLS (Multi-Protocol Label Switching/Generalized Multi-Protocol Label Switching) network, virtual private networks are provisioned as Label Switched Paths tunnels. Current virtual networking access and control methods are based on statistical multiplexing of simple conceptual models such as virtual links, virtual switches or virtual routers. These simplistic models have the advantage of keeping the complexity and the details of the underlying infrastructure invisible (and inaccessible) to customers, and instead provide them with a uniform, simple abstraction as the interface.

C) Virtual Machine Technology

Since 2005, virtual machines are again becoming a hot topic in academia and industry. Venture capital firms are competing to fund startup companies touting virtual machine based

technologies while major hardware manufacturers Intel, AMD, Sun Microsystems and IBM are developing virtualization strategies that target markets with revenues in billions and growing. In research labs and universities worldwide, researchers are developing approaches based on virtual machines to solve mobility, security and manageability problems [12].

Some of the many benefits of using virtual machines are: simplicity of service management, application scheduling flexibility, ease of testing and debugging complex systems, hardware multiplexing, enhanced system management capabilities, the ability to run conflicting processes in isolation from each other, heightened system security, and live migration, dynamic content distribution and secure trusted computing.

3. CLOUD SERVICES

Clouds are also usually categorized in three types [2], depending on the abstraction level of the provided services.

- Infrastructure-as-a-Service (IaaS) clouds supply virtual hardware resources such as machines, networks, or storage. IaaS is what the user should opt from virtual computers, cloud storage, network infrastructure components such as firewalls and configuration services. Usage fees are calculated per CPU hour, data GB stored per hour, network bandwidth consumed, network infrastructure used per hour, value added services used, e.g., monitoring, auto-scaling etc. A well-known example is Amazon EC2. Here, users must adapt, deploy, and control the software stack their applications rely on.

- Platform-as-a-Service (PaaS) clouds provide a container environment for users to run their software components— for instance, an application container, an online database management system shared with several users, other mechanisms for data persistence to be used by online applications deployed over the platform, etc. PaaS offerings facilitate deployment of applications without the cost and complexity of buying and managing the underlying hardware and software and provisioning hosting capabilities, providing all of the facilities required to support the complete life-cycle of building and delivering web applications and services entirely available from the Internet. Examples are Google App Engine (GAE) and Microsoft's Azure.

- Software-as-a-Service (SaaS) clouds give access to typical applications (e-mail clients, etc.) through the network. As such, they are intended for final users, not for system developers. In SaaS (Software as a Service), the consumer is free of any worries and hassles related to the service. The Service Provider has very high administrative control on the application and is responsible for update, deployment, maintenance and security. The problem of dealing with education-supporting technologies is here reduced to selecting and aggregating in an appropriate order a set of prepackaged software services.

4. VIRTUALIZATION

Virtualization was first developed in 1960's by IBM Corporation, originally to partition large mainframe computer into several logical instances and to run on single physical mainframe hardware as the host. This feature was invented because maintaining the larger mainframe computers became cumbersome. By day to day development, virtualization technologies have rapidly attains popularity in computing; in fact it is now proven to be a fundamental building block for today's computing [9].

Two primary benefits offered by any virtualization technology are 1.Resource sharing and 2.Isolation. Resource sharing - Unlike in non-virtualized environment where all the resources are dedicated to the running programs, in virtualized

environment the VMs shares the physical resources such as memory, disk and network devices of the underlying host. The resources are allocated to the virtual machine on request. Hypervisors plays a significant role in resource allocation. Isolation - One of the key issue in virtualization, provides isolation between virtual machines that are running on the same physical hardware. Programs running in one virtual machine cannot see programs running in another virtual machine. This is contrast to non-virtual environment where the running programs can see each other and if allowed can communicate with each other.

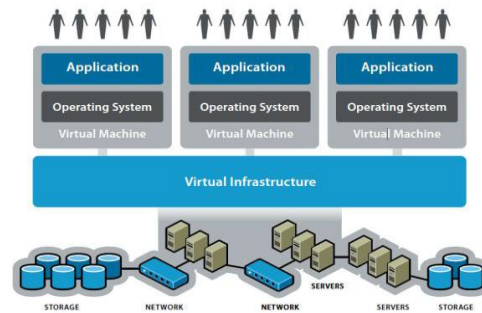


Figure- 1 Virtualization System Architecture

Above figure presents the Virtualization System Architecture. System architecture clearly shows that Virtualization allows multiple OSEs to share a single physical interface, to maximize the utilization of computer system resources, such as I/O devices. A virtual machine is a tightly isolated software container that can run its own operating systems and applications as if it were a physical computer. A virtual machine behaves exactly like a physical computer and contains its own virtual CPU, RAM, hard disk, and network interface card (NIC). An operating system can't tell the difference between a virtual machine and a physical machine, nor can applications or other computers on a network. Even the virtual machine thinks it is a real computer. Nevertheless, a virtual machine is composed entirely of software and contains no hardware components whatsoever. As a result, virtual machines offer a number of distinct advantages over physical hardware. An additional software layer, named Virtual Machine Monitor (VMM) or hypervisor [3], is introduced to provide the illusion of Virtual Machines (VMs), on top of which each OS assumes owning resources exclusively. Mainly there are two kinds of virtualization. They are: Full virtualization and partial virtualization.

A) Full Virtualization:

In this approach the hypervisor simulates several logical instances of completely independent virtual computers possessing their own virtual resources. These virtual resources include IO ports and DMA channels. Therefore, each virtual machine can run any operating system supported by the underlying hardware. The virtual machine environment that provides "enough representation of the underlying hardware to allow guest operating systems to run without modification can be considered to provide "Full Virtualization" [5]."

B) Partial virtualization:

Unlike full virtualization, in partial virtualization the running guest OS should be modified in order to be operated in the virtual environment. Partial virtualization is a subset of server virtualization, which provides a thin software interface between the host hardware and the modified guest OS. One of

the main characteristics of partial virtualization technology is, the virtual machine monitor is simple which allows partial virtualization to achieve performance closer to nonvirtualized hardware. Device interaction in partial virtualized environment is very similar to the device interaction in full virtualized environment.

5. MASHUP CONTAINER

The Mashup and the Cloud Computing worlds are strictly related because very often the services combined to create new Mashups follow the SaaS model and, more in general, rely on Cloud systems. Moreover even the Mashup platforms may rely on Cloud Computing systems as already happens for IBM Mashup Center [8] and JackBe Enterprise Mashup Server [6]. Mashup can be of two types: Data Mashup and Event Driven Mashup.

Data Mashups, i.e., those Mashups that combine data extracted by different sources, e.g., Yahoo Pipes! [7]. Event Driven Mashups, i.e., those Mashups in which the basic components - called Services (Svc) from now on-interact through events rather than through the classical Call-Response paradigm. Event Driven Mashups are typically entered through a graphical editing tool which is part of a platform called Service Creation Environment (SCE). The Mashup creator drags and drops blocks corresponding to Svcs and draws edges corresponding to the dependencies among them.

A) Requirements

- Deployment support, for new Mashup deployment according to the PaaS approach.
- Scalability support, for the simultaneous execution of a very large number of Sessions. The number of Mashups available in the Mashup Container and the number of Mashups that are actually in execution at a given time are supposed to be high and growing as new Mashups are continuously deployed /executed in the system.
- Fault Tolerance, to ensure that even if one or more components fail, the running Sessions continue their execution.
- Low latency and high throughput, to effectively support the execution of fine-grain composite service execution. Latency is very important.
- Authentication, Authorization and Accounting (AAA) support, to allow for a seamless integration of the Mashup with the AAA modules already existing in the owner platform.
- Management support, to have the complete control of the platform resources, to control Mashup activation / execution as well as to allow the platform administrator to perform appropriate management actions.

B) Mashup Container Management

The Mashup Container is supposed to provide some mechanisms to support the management of Mashups usage. Security is probably the most important aspect. Security must be intended in the following two ways. The first is related to the protection of the system from external malicious service providers while the second is related to the management of user roles. This second aspect is very important for Enterprise Mashups in which different employees probably have different rights in accessing company data. The Mashup Container provides a set of statistics that allow the administrator to have a clear picture of what is happening inside the platform. The Mashup Container supports the deployment and execution of Event Driven Mashups. New composite Services are developed by means of a graphical tool and deployed in the container to be used by end users. The Mashup Container can be located—in the Cloud and

made available to the user according to the Platform as a Service model. The Mashup developers may deploy Mashups in the provider platform without taking care of low level issues as network support, memory/disk size and CPU performance. In order to achieve this objective, the system takes advantage of the functionalities provided by virtualized environments [4].

6. PERFORMANCE

Our approach takes into account the configuration of hosts and the time varying demands of workloads, i.e. resource usage traces of the application over time. The costs-per-host include the host list price, license and maintenance fees for a virtualization solution, and host power usage. The time varying demands of workloads are customer specific. We assume a three year lifetime for the hosts. Initially, a desirable host configuration is chosen for the resource pool. The host has a certain capacity in terms of processing CPU cores and memory. An automated consolidation exercise packs the workloads to a small number of these hosts. The approach takes into account the aggregate time varying (multiple) resource usage of the workloads and a given capacity of the hosts. Multiple host alternatives can be considered iteratively. later, we apportion the cost of the shared hosts in the pool among the hosted workloads. If the cost associated with a workload is greater than the cost of a smaller server that could also host the workload, then the workload is a candidate for right-virtualizing. The method can be repeated for different combinations of resource pool host and smaller server host configurations. Finally, we evaluate the average resource usage in the pool to make sure that the selected host configuration for the resource pool is balanced and well utilized.

7. CONCLUSION

Cloud computing is a new infrastructure deployment environment that delivers on the promise of supporting on-demand services like computation, software and data access, storage etc. Clouds provide inexpensive access to remote resources. In this paper, Virtualization and virtual machine environment is implemented. Virtualization brings very little added security to the environment. Virtualization is a powerful solution to reduce the operational costs in today's computing but if done wrong it become as a threat to the environment. While implementing, exaggerate the security model to with stand the attacks. We implement the virtualization technology with virtualized infrastructure architecture and effectiveness of PaaS based Event Driven Mashup container using virtualization by measuring CPU and memory usage in shared resource environments. we use the architecture and the organization of a Mashup Container that supports the deployment and the execution of Event Driven Mashups. In collaboration with PaaS, Virtualization provides an opportunity for extension of independent virtual resources based on available physical systems.

8. REFERENCES

- [1] Onur, E., Sfakianakis, E., Papagianni, C., Karagiannis, G., Kontos, T., Niemegeers, I., Chochliouros, I.P., de Groot, S.H., Sjodin, P., Hidell, M., Cinkler, T., Maliosz, M., Kaklamani, D.I., Carapinha, J., Belesioti, M., Fytros, E., Fac. of Electr. Eng., Math. & Comput. Sci., Delft Univ. of Technol., Delft, Netherlands, —Intelligent End-To-End Resource Virtualization Using Service Oriented Architecture, GLOBECOM Workshops, IEEE, 28 December 2009 .

- [2] Luis M. Vaquero, Member, IEEE, “EduCloud: PaaS versus IaaS Cloud Usage for an Advanced Computer Science Course”.
- [3] VMware. VMware security center. <http://www.vmware.com/support/security.html>.
- [4] R. Uhlig, G. Neiger, D. Rodgers, A. L. Santoni, F.C. M. Martins, A.V. Anderson, S. M. Bennett, A. Kagi, F. H. Leung, and L. Smith, —Intel Virtualization Technology, IEEE Computer Volume 38, Issue 5, pp. 48–56, May 2005.
- [5] J. Kirch. Virtual machine security guidelines. The center for Internet Security, September 2007. http://www.cisecurity.org/tools2/vm/CIS_VM_Benchmark_v1.0.pdf.
- [6] Presto Mashup Server, www.jackbe.com
- [7] Yahoo Pipes!, <http://pipes.yahoo.com/pipes/>
- [8] IBM MashupCenter, www.ibm.com/software/info/mashupcenter/
- [9] M. Rosenblum, T. Garfinkel. —Virtual Machine Monitors: Current technology and future trends, Computer, 38(5), Los Alamitos, CA, IEEE Computer Society Press, pp. 39-47, 2005 .
- [10] The Open Group, “Service Oriented Infrastructure Reference Framework”, <http://www.opengroup.org/projects/soasoi/uploads/40/19218/soi-V1-5-P1.pdf> , 2009
- [11] Khanvilkar, S., Khokhar, A., "Virtual Private Networks: An Overview with Performance Evaluation", IEEE Communication magazine, vol. 42 (10), pp. 146-154, October 2004
- [12] M. Rosenblum and T. Garfinkel. Virtual Machine Monitors: Current Technology and Future Trends. IEEE Computer, 35(5):39-47, 2005