Dynamic Resource Allocation Using Virtual Machines for Cloud Computing Environment

OBJECTIVE

we present a system that uses virtualization technology to allocate data center resources dynamically based on application demands and support green computing by optimizing the number of servers in use.

DOMAIN:

Cloud Computing

SYNOPSIS

we issue to achieve the goal of management multiple virtualization platforms and multiple virtual machine migrations across physical machines without disruption method. We discuss that ensure load balance when multiple virtual machines run on multiple physical machines. we present a system which is implementation of optimization with Dynamic Resource Allocation dealing with virtualization machines on physical machines, practice DRA method in this system. The dynamic results confirmed that the virtual machine which loading becomes too high it will automatically migrated to another low loading physical machine without service interrupt. And let total physical machine loading reaching balance.

It is however unclear whether this technique is suitable for the problem at hand and what the performance implications of its use are. We found out that this approach results in a tractable solution for scheduling applications in the public cloud. In the same method becomes much less feasible in a hybrid cloud setting due to very high solve time variances. In the cloud model is expected to make such practice unnecessary by offering automatic scale up and down in response to load variation. It also saves on electricity which contributes to a significant portion of the operational expenses in large data centres. We develop a set of heuristics that prevent overload in the system effectively while saving energy used. It trace driven simulation and experiment results demonstrate that our algorithm achieves good performance.

EXISTING SYSTEM

Existing works on autonomic management systems for virtualized server environments tackle the allocation and placement of virtual servers from different perspectives. Virtual machine monitors (VMMs) like Xen provide a mechanism for mapping virtual machines (VMs) to physical resources. Mapping is largely hidden from the cloud users. Users with the Amazon EC2 service Example do not know where their VM instances runs. It's up to the cloud provider to make sure the underlying physical machines (PMs) have sufficient resources to meet their needs.

LIMITATIONS

- A policy issue remains as how to decide the mapping adaptively so that the resource demands of VMs are met while the number of PMs used is minimized.
- No control over the business assets (data!). The main assets in every company are its data files with valuable customer information.
- Risk of data loss due to improper backups or system failure in the virtualized environment.
- High cost and loss of control.

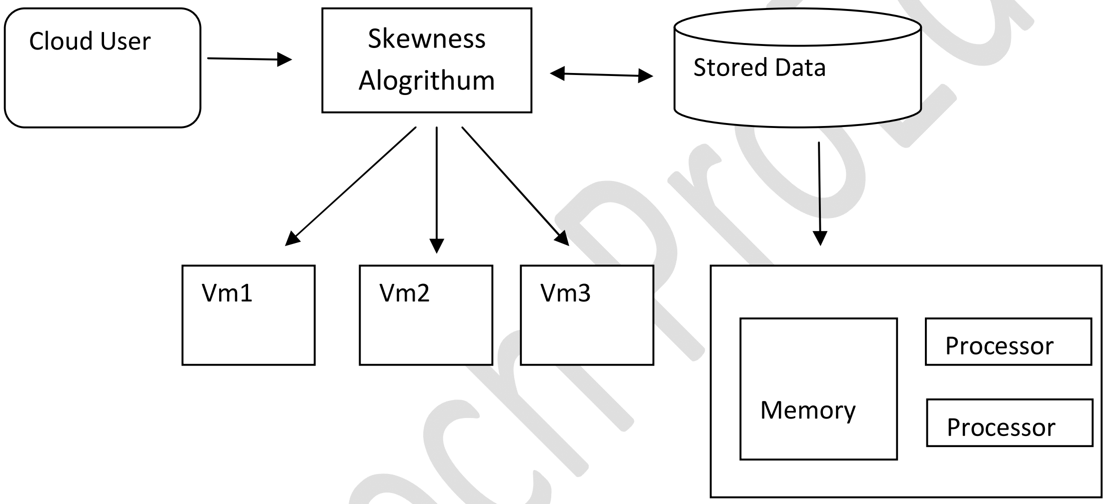
PROPOSED SYSTEM

We present the design and implementation of an automated resource management system that achieves a good balance between the two goals. We make the following contributions. Overload avoidance: The capacity of a PM should be sufficient to satisfy the resource needs of all VMs running on it. Otherwise, the PM is overloaded and can lead to degraded performance of its VMs. Green computing: The number of PMs used should be minimized as long as they can still satisfy the needs of all VMs. Idle PMs can be turned off to save energy. We develop a resource allocation system that can avoid overload in the system effectively while minimizing the number of servers used. We introduce the concept of "skewness" to measure the uneven utilization of a server. By minimizing skewness, we can improve the overall utilization of servers in the face of multidimensional resource constraints. We are using cloudsim for implementations.

ADVANTAGES

- ➢ A flexible, scalable infrastructure management platform has been architected and a prototype implemented
- ➢ Measurement of resource usage and end user activities lies hands of the cloud service provider.
- ➢ Opaque cost structure due to highly flexible usage of cloud services.
- ➢ Stable of cost structure.

SYSTEM ARCHITECTURE



HARDWARE SPECIFICATION

- Main Processor : 2GHz
- Ram : 512 MB (min)
- Hard Disk : 80 GB

SOFTWARE SPECIFICATION

- Language : Java
- Web Server : Tomcat 6
- Operating System : Windows 7 32 Bit
- CloudSim