Semestrial report

The challenges in building a scalable virtualized datacenter management: a survey on the state of the art

2012/2013-1

Peter OROVA

E6P9SA

peter.orova@gmail.com

1 Introduction

Virtualization has been an ever spreading phenomenon in the IT world for the past few years. Various vendors have created their own terminology, methodology, and array of products related to this specific technology. It is essential to understand however, that there exist a number of fundamentally different types of virtualization, each with different structure and objective. This work focuses essentially on data center grade virtualization that mainly uses platform virtualization[1].

The motivation of virtualization in datacenter grade environment is twofold. On the one hand, an ever increasing need for efficiency hence reduced operational costs. On the other, virtualization enables the transition of legacy softwares onto new hardware infrastructure.

Virtualization changes some fundamental aspects of system management, resolving difficulties on the one hand, and introducing challenges on the other. The management of physical hardware in a virtualized environment still requires effort, albeit a slightly different kind than in cases, where no virtualization is present. Additionally, resulting from the virtualized nature of the infrastructure, virtualization specific management tasks emerge.

The challenge further escalates, when virtualization is applied in a large scale infrastructure of several hundreds or thousands hosts. System management and also corporate leadership must be made aware of the fact, that although virtualization is a powerful tool in some cases, it is far from being the proverbial 'silver bullet'.

In the following, I examine the general impacts of virtualization on system management, and the requirements that emerge from the scaling. In the second half of this work, some of VMware's solutions will be presented to the various challenges posed by the management of virtualized datacenters.

[1] For an overview on virtualization technologies see [1]

2 Data center management

2.1 Introduction

In a datacenter environment the main goal of virtualization is to enable higher resource utilization of the available resources. Coincidentally, virtualization renders IT provisioning much easier and cheaper. This ease and cheapness of provisioning causes the data centers to boom in size. The datacenters with several thousand virtual machines pose a great challenge for system administrators. To help manage such an infrastructure, virtual system solution vendors such as VMware provide specialized administration tools. The requirements that these administration tools have to fulfill however are daunting: high availability, consistency, robustness, backward compatibility, and scaling. At the same time system management is not permitted to consume many resources.

From a bird view, the structure of a virtualized compared to a non-virtualized can is the following:

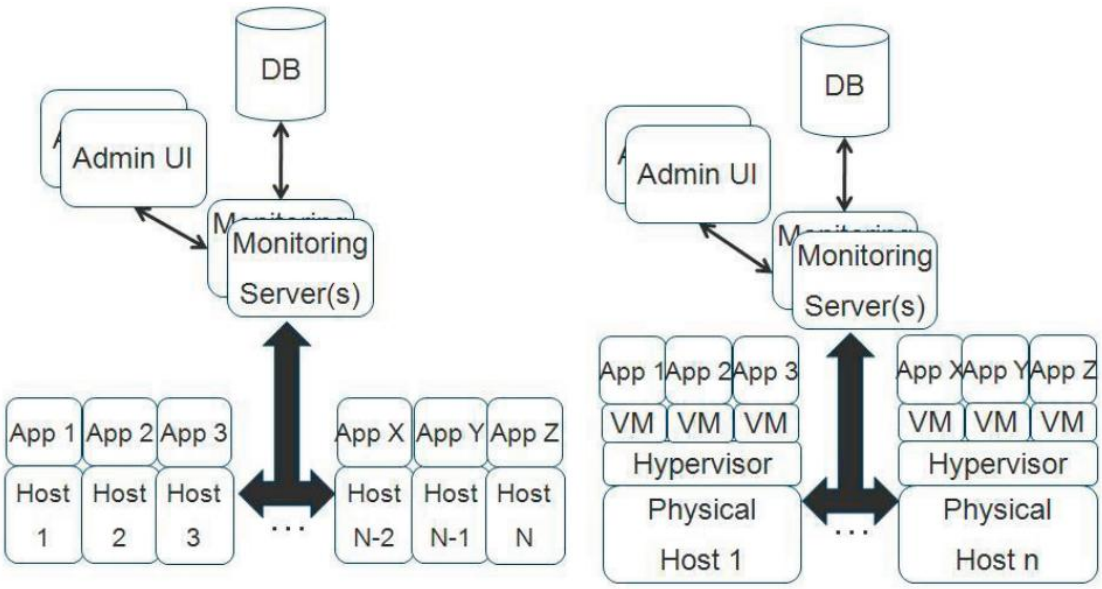

Figure 1. : Physical (left) and Virtualized (right) infrastructure[2]

[2] Source: [2], page 96.

Figure 1 shows the difference between the two types of approaches. In the virtualized case, the monitoring servers may collect data from both physical hosts, and hypervisors. In the case of the non virtualized approach, data collection must be performed on a per physical host basis. It is visible that in the virtualized case, the applications run on virtual machines instead of physical ones. This affects the nature of the management tasks greatly.

This section discusses the management tasks in a data center, comparing the nature of those in a non-virtualized and a virtualized environment. Subsequently, the requirements emerging from scaling the datacenter are reviewed.

2.2 Management tasks

The following tasks of system management have been compiled in [2].

2.2.1 New system deployment

Amongst the basic tasks of system management, new system deployment is the first. In a physical infrastructure, this task involves the installation of new hardware, together with the necessary software components. Constraints to respect are: the available power of the given environment.

In a virtualized environment however, deploying a new system involves the creation of a virtual machine, and the installation of the necessary software components on it. Two types of impact of the virtualized environment can be observed here: the deployment itself can be performed solely on the logical layer, thus all the hassle with the installation of new hardware is avoided. The other impact is, that the same software components are installed on the virtual machine as the ones used for a physical machine.

2.2.2 Application deployment

Application deployment in physical and virtualized infrastructures is not very different in the usual case. In case, the application to be deployed has special requirements, such as the need to run it on a separate machine possibly with specific performance constraints, the deployment process differs in ways discussed in 2.2.1.

2.2.3 Planned maintenance

One of the difficulties of planned maintenance is to provide minimal or ideally optimal service levels even during the maintenance window. In a physical environment, this requires the reconfiguration of several hardware and software components. In a virtualized environment however, the virtual machines can be ‘evacuated’ from the physical host that needs maintenance. Moving virtual machines is a network intensive, but otherwise not complicated task. The benefits of the virtualized infrastructure for planned maintenance are the following: there is no need to reconfigure hardware components or software components. On the other hand, it is necessary to be able to move the virtual machines from one physical host to the other. Overall, planned maintenance in a virtualized infrastructure can be performed easily in such a manner, that the level of service provided does not deteriorate.

2.2.4 Performance monitoring and debugging

In case of performance complaints from end users, in a physical infrastructure, administrators must inspect both hardware and software components of the system in question. Inspecting hardware can mean that the administrator in fact needs to locate and ‘visit’ the hardware. In a virtualized environment, the system administrator can determine the source of the problem solely by inspecting the logical layer of the infrastructure. This is both cheaper and more easily achieved. It must be noted however, that virtualized environments may be more exposed to hardware faults, as multiple virtual machines may run on the same hardware.

2.2.5 Backup and recovery

Backup and recovery operations are quite similar in both cases, except the Snapshot operations, introduced by virtualization, and discussed in 2.2.7.

2.2.6 Live migration

Live migration is a virtualization enabled infrastructure specific management task. It means to move a virtual machine from one (physical) host to another, while it is running. This capability becomes very useful in two main scenarios: since the virtual machines can be moved from one physical host to the other, performance considerations can be taken into account. In other words, virtual machines from a host under heavy load can be moved to a host with free capacity. On the other hand, live migration can be

used to consolidate low resource demand virtual machines running on different physical hosts, to save power on the unneeded hosts. The other scenario where live migration comes in handy is planned maintenance. The virtual machines from the physical host that are scheduled for maintenance can be evacuated to ensure continuous service, while enabling maintenance operations.

2.2.7 Snapshot operations

Snapshot operations in this context are tasks that are specific to virtualization enabled infrastructures. Usually there exist three different types of snapshot operations.

- **Create snapshot**: the user can create a 'snapshot' i.e. an image of the current state of the system. Until the snapshot is available, the user can 'reload' it to nullify unwanted changes that occurred after the creation of the snapshot.
- **Revert to snapshot**: is the action of 'reloading' the snapshot. A typical use case of this operation is the following: user creates a snapshot, and then installs a new application or patch. If the newly installed software components cause system instability, the user can choose to revert to the previously saved snapshot thus annihilating all negative effects of the newly installed software.
- **Commit snapshot:** is the action where the actual snapshot is to finalize the state of the virtual machine, changing its state definitely and removing the snapshot.

It is interesting to note, that from a certain perspective, snapshot operations can be considered to be in the scope of system management, as they indeed are a form of backup and recovery. However, as mentioned above, snapshot operations can be requested by the end user.

2.2.8 Cloning

The final operation that is discussed here is also specific to virtualized infrastructures. Cloning means to create a copy of an existing off-line virtual machine. A typical use case for cloning: new virtual machine is needed so it is created based on a master virtual machine with the help of cloning.

2.2.9 Conclusion on management tasks

In 2.2.1 through 2.2.7 some of the basic management tasks have been discussed in a very coarse detail. The distinction between the fundamental nature of the tasks has been

highlighted, and additionally, virtualized infrastructure specific tasks have also been mentioned. In general, it is safe to state based on [1] and [2], that system management is greatly simplified by the introduction of data center level virtualization.

2.3 Requirements emerging from scale

2.3.1 Introduction

The management tasks that the system administrators must be able to perform have been discussed in 2.2. This section examines the requirements posed against the management and virtualization layer of an infrastructure that scales greatly.

2.3.2 Fairness

The virtualization layer must provide a fair share of the resources to the various users and administrators. 'Fair share' here is understood, as the share of resources that the given user paid for. This fairness must be ensured, even in cases when the physical machines are 'overbooked' i.e. more virtual resources are allocated than physically exist. A large scale infrastructure, be it virtualized or not, is managed by several (hundred) administrators, depending mainly on the size. A fairness of resource allocation must also be respected with regards to the administrators.

2.3.3 Security

Three main aspects of security arise in a large scale virtualized infrastructure:

- **Virtual machine interference:** users must not be able to interact with each other's virtual machines. Furthermore, the users must not be able to use covert channels to inspect infrastructure elements that they are not supposed to interfere with.
- **Starvation:** in a close connection to fairness, a virtual machine must not be allowed to consume all of the resources of a physical host, thus causing other user's virtual machines to starve.
- **Authorization:** the management and virtualization layer must be able to manage the authorizations on different elements of the infrastructure. For each entity in the infrastructure, a set of users together with their respective authorized operations must be defined. The complexity of the authorization booms with the scale of the system, since

the possible <user, entity, operation> 3-tuples multiply with each added user and infrastructure element.

2.3.4 Robustness

The nature of the infrastructure in different environments can vary greatly. A good example for this is the contrast between a geographically dispersed large company and a small to medium businesses. In the first case, high latency, low bandwidth connections can pose a performance bottleneck, and the large number of hosts may call for automation. In the second case only a relatively small number of manually manageable hosts is present, and those are connected with high bandwidth low latency lines. The virtualization, and also the management layer must be constructed in such a way, that the necessary operations can be performed with relative ease, regardless of the nature of the infrastructure.

2.3.5 Availability

As more and more enterprises place mission critical applications in virtualized infrastructure, the virtualization and management layer in fact may are also classified as mission critical resources. Therefore it is imperative that virtualization and management layers be able to provide performance and availability guarantees. Availability of the management layer is all the more important when the physical infrastructure is in a different location than the system administrators.

2.3.6 Compatibility

Especially in large scale infrastructures, the updates on the management layer software do not happen simultaneously. Therefore, the management layer must be able to cope with scenarios, where multiple versions are simultaneously deployed. The main issue in such scenarios is backwards compatibility.

2.3.7 Conclusion on requirements

In this section the main, high level requirements posed against the virtualization and management layer have been discussed. It is intuitively visible, that fulfillment of these requirements becomes greatly challenging at scale, due to a number of factors, ranging from the sheer number of infrastructure elements, to the available physical connections in a multi site infrastructure.

3 VMware approaches

3.1 Introduction

This section examines the solutions that VMware provides to the various requirements outlined in 2.3. The examined software solution is VMware vSphere as indicated in [2]. The architecture of vSphere is very similar to the virtualized datacenter architecture shown on Figure 1.

3.2 Overview of VMware specific solutions

The following section is compiled based on [2]

3.2.1 Fairness

Fairness of resource allocation is ensured by two entities in the vSphere architecture. At ESX level, schedulers are responsible for the proper allocation of resources for each ESX host. DRS (Distributed Resource Scheduler) works together with the ESX scheduler.

An important aspect of fairness is the fairness of management operations. Management operations often are resource intensive, therefore the appropriate allocation is not trivial. Some examples of resource intensive management operations are frequently powering on and off virtual machines, or performing a large number of live migrations simultaneously.

VMware's solution for this problem is twofold: there exists a configurable limit for the amount of a specific management tasks per host, per network and per storage. On the other hand, each user and session has a waiting queue, and scheduling ensures that no user starves.

3.2.2 Security

The issues of security discussed in Security 2.3.3 are addressed in vSphere in the following manner. Isolation of virtual machines is ensured by the ESX hypervisor itself. To manage the dangers emerging from network communications vShield Zones provides firewall, NAT, and intrusion detection mechanisms. Permission handling is

provided by vSphere in the form of 3-tuples: <user, action, object>. This security model works well in small scale systems but can become hopelessly complex with scale.

3.2.3 Robustness

The issues of robustness primarily occur in geographically distributed large systems. The main issue is the high latency low bandwidth connections between the sites, which cause both administrators and users to have a less than optimal access to the infrastructure. Moreover, links may at times become entirely unavailable.

The issue of bandwidth is dealt with by compressing all data that goes through the network, be it configuration changes or live migrations. Additionally, clients can subscribe to fine grained vCenter Server changes, which makes the sending of whole configuration changes unnecessary.

3.2.4 Availability

Availability is a key factor in datacenters. Physical components will inevitably fail with some frequency, therefore the infrastructure must be designed in such a way to tolerate these outages.

VMware proposes VMware Fault Tolerance, which duplicates the active virtual machine with one that runs in lock step with that. In case of failure of the active virtual machine, the backup can immediately take its place.

On a larger scale, to limit the effect of failing vCenter monitoring servers, multiple vCenter monitoring servers are linked together in such a way that they fail independently. Another approach used by VMware is to keep a hot standby for the vCenter monitoring application.

3.2.5 Backwards compatibility

VMware ensures that all communications between the vCenter Server and all the hosts is versioned. Moreover, each vCenter Server version has a communication agent that it can upload to each host, to ensure compatibility and performance.

Another issue with compatibility is the vSphere management API. Certain aspects of older versions of the management API can cause performance problems, but on the other hand, many third party vendors rely on them for their products. It is therefore a slow process to phase out the deprecated management API.

3.3 vSphere Data Protection

3.3.1 Introduction

Amongst the tasks of a system administrator, one of the most important is backup and recovery. Numerous tools exist to help to streamline this process. VMware's solution is vSphere Data Protection (VDP) [3]. The main advantage of VDP is, that it is fully integrated with vCenter Server, therefore it makes a reasonable choice of backup and recovery tool for that environment.

3.3.2 Feature highlights

As mentioned before, VDP is integrated with vCenter Server, therefore management of VDP is performed using vSphere Web Client. VDP uses proprietary technology that enables deduplication of the data, thus rendering disk space use more efficient. VDP comes in an appliance form, that, and the backups use a checkpoint and rollback mechanism for enhanced protection.

3.3.3 Architecture

To ease the deployment of VDP, it comes as a preconfigured Linux-based appliance. Each virtual appliance is configured to support the backup of 100 virtual machines at most. One vCenter Server can run up to 10 VDP appliances.

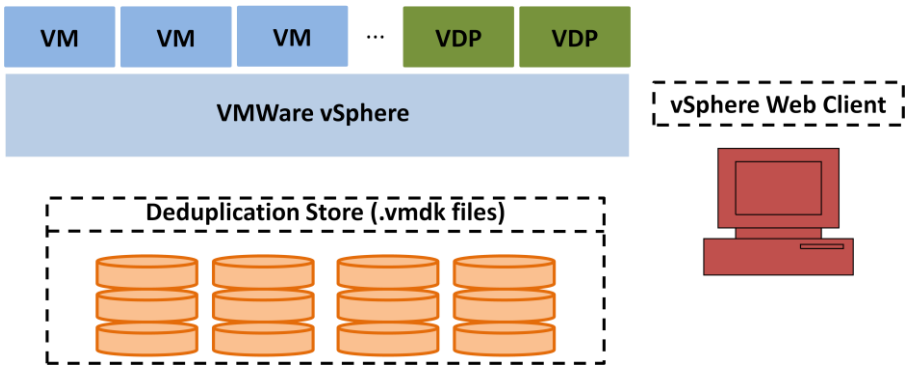

Figure 2. VDP architecture[3]

[3] Source: [3], page 4.

3.3.4 Use

Each virtual appliance comes with preconfigured storage sizes. Appliances are packaged using the .ova (Open Virtualization Archive) format.

3.3.4.1 Installation

After deployment, initial configuration such as networking and time information. The reboot after the initial configuration is lengthy, it can take up to 30 minutes to finish. After the initial installation, VDP creates the first backups of the virtual machines it is configured to protect. This, due to the fact that any subsequent backups use CBT (Changed Block Tracking) and deduplication, takes a relatively long time.

3.3.4.2 Management

- **Backup policy:** the period of retention of the backups of the protected machines can be specified in the following manners: until forever, for a fixed period of time, until a specific date. The last option is a kind of 'rolling' retention policy e.g. daily backups with 15 days retention periods.

- **Restore:** Two types of restore exist: a virtual machine restore and individual file/folder restore. The former is performed by the administrator using the vSphere Web Client. Upon a restore request, VDP estimates the load of a full restore, and one, that uses CBT. In environments where the rate of change on the storage is high, it is usually better to do a full restore, instead of calculating the increments using CBT. VDP chooses whichever method that requires the least resource. The newly restored virtual machine can be assigned to a different data store to prevent overwriting data. The file/folder level restore is performed using vSphere Data Protection Restore Client, and can be performed by end users.

- **Checkpoint/Rollback of VDP itself** prevents backup data corruption. This mechanism protects the backup data, by checkpointing the VDP appliance itself. Naturally, in case of failure, jobs performed after the last checkpoint would be lost, but the backup data of the VDP protected virtual machines will not be harmed.

3.4 Distributed resource management

3.4.1 Introduction

One of the objectives of infrastructure management is to provide the required services using optimal resource consumption. In larger scales, it may occur that the resource demands of services increase in one part of the infrastructure, while in other parts it decreases. For these cases load balancing measures must be taken to ensure optimal resource utilization. However, the possible impacts of load balancing are numerous, therefore one must be very careful when taking these measures. In large scale infrastructures it is not practical to entrust this task to human operators. Resource scheduling and allocation must take place at least party autonomously, with the supervision of system administrators.

VMware's solution for large scale resource allocation and scheduling is the VMware Distributed Resource Management (DRS) [4]. In the following, the basic aspects of DRS will be discussed.

3.4.2 Basic resource controls

To be able to speak about resource allocation, first a resource model must be established. DRS uses a resource model is constructed in such a way that resource allocations can be given in both absolute and relative manner. The terms and their definitions that DRS uses are as follows.

- **Reservation** is the minimal amount of resource that is guaranteed to a given instance. Reservation is expressed in absolute values such as MHz and MB.

- **Limit** is the maximal amount of resources that a given instance may be allocated. This upper limit is valid, even if free resources are available, and theoretically could be allocated to the given instance.

- **Share** expresses the relative importance of the instance in terms of resource allocation. A VM minimally is allocated a portion of the resources according to the ratio of his 'share' number and the total 'shares' on the given resource. The 'share' can be thought of as relative 'weight'.

3.4.3 Resource pools structure

System administrators may set the reservation, limit and share values on a per VM basis. However, it is trivial, that even in cases where only a few hundred virtual machines are present, this task is overwhelming.

VMware introduced therefore the "Resource pool" abstraction. In fact, a resource pool is a group of virtual machines. Resource pools may contain virtual machines or other sub resource pools. The resulting structure of the pools and VMs is a hierarchy. This hierarchical structure often reflects the internal structure of the organization it serves.

The resource pools have the same 3-tuple for resource allocation: reservation, limit and share. The rule of admission into a pool is that the sum of the reservations of the elements in the pool cannot exceed the reservation of the pool itself.

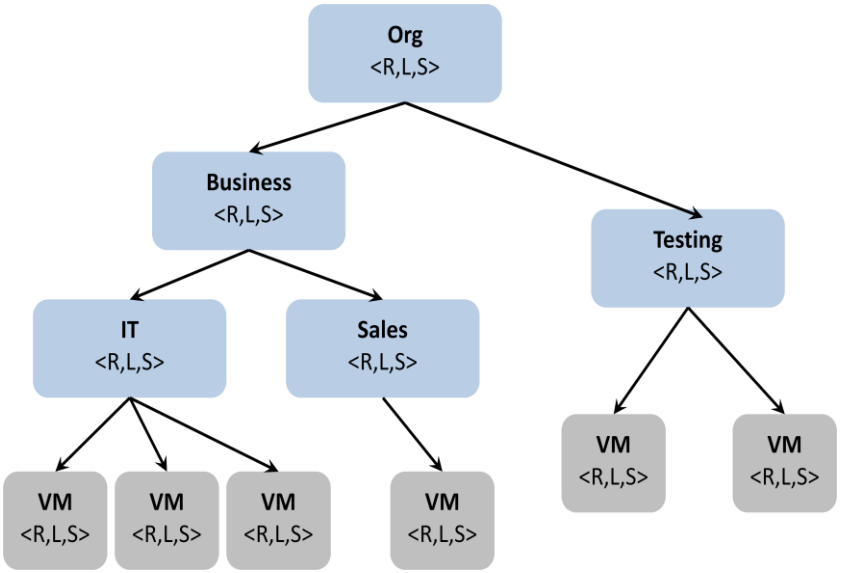

Figure 3. An example of resource pool hierarchy[4]

The root resource pool – in the example of Figure 3, "Org" – represents the totality of the physically available resources.

3.4.4 Divvy algorithm

The resource allocation in a resource pool tree in fact consists in the repartition of the available resources of the root node to its children, and that, in a recursive manner. In the DRS resource pool tree, the algorithm that is responsible for both CPU and memory

[4] Source: [4],page 47.

allocation is called “divvy”. Three types of divvy algorithms exist: reservation-divvy, limit-divvy and share divvy, for the calculation of the respective reservations, limits and shares.

The divvy algorithm works in two phases. First a bottom-up phase gathers the demands for the available resources. For the CPU, the demand is calculated in the following manner:

$$CPU_{demand} = CPU_{used} + CPUrun / (CPU_{run} + CPU_{sleep}) * CPU_{ready}$$

In other words, CPU demand is the sum of the actual CPU usage and the scaled portion of the time it was ready, but was not allocated actual CPU time. Memory demand is determined by the sampling the activity on a randomly selected set of memory pages during a period of time.

When all the demand values are set, they are collected up the tree with an adjustment: demands can nor exceed the limit on the given VM or resource pool, neither be inferior to the reservation of the respective element.

The second phase of the divvying algorithm is performed top-down. Using the values of reservation, limit and demand the allocations are performed obeying the following 3 rules

- Each child is allocated the resources proportionally to its share
- The allocated amount of resource is at least the reservation of the given node
- The allocated amount of resource is at most the limit of the given node

The limit values of the child nodes are modified according to the following formula:

$$limit_i = \min(limit_i, demand_i)$$

3.4.5 Load balancing

3.4.5.1 Load balancing algorithm

The load balancing algorithm uses a locally optimal hill climbing technique. This approach usually gives a globally sub optimal solution, but has the advantage of being practical in such dynamically changing environment. The objective of the DRS algorithm is to “*minimize the cluster wide imbalance: I_c*” [4]. The aforementioned hill climbing technique in this case consists in selecting a single virtual machine migration

that, if performed, reduces the imbalance the most. The selection of 'moves' continues until no improving moves can be made in a reasonable threshold. The algorithm taken from [4] is as follows:

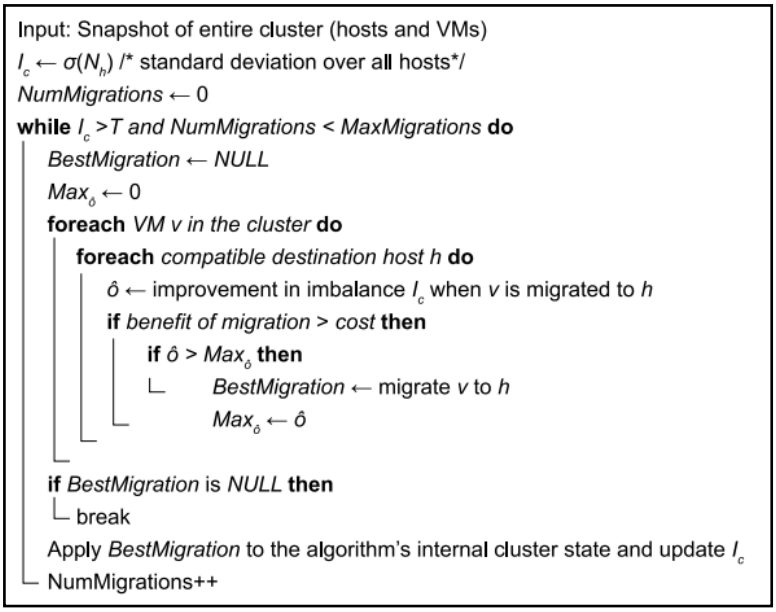

Figure 4. The load balancing algorithm used by DRS[5]

According to the authors, the algorithm shown in Figure 4 has been greatly simplified in order to highlight the nature of the load balancing mechanism. The real algorithm observes many more factors, some of which are discussed in the following sub-section.

3.4.5.2 Load balancing considerations

This section examines very briefly some other factors that the load balancing algorithm outlined on Figure 4 uses.

- **Minimum goodness:** the selected 'most beneficial' VM displacement is not performed, in case it does not cross a 'minimum goodness' threshold. The threshold for this rule is dynamically calculated taking into consideration the number of hosts and virtual machines.

- **Cost-Benefit analysis**: The moving of the virtual machines with vMotion has a non trivial cost. For a VM that is under a memory heavy use, moving creates additional necessary memory operations. The disruption of service on the given VM caused by vMotion moving the machines must also be taken into consideration.

[5] Source: [4], page 50.

- **Affinity rules:** DRS supports the enforcement of VM-to-VM and VM-to-host rules. VM-to-VM *affinity* rules give a set of VMs that must be kept on a single host. A possible use case for this, is when the two VMs run in lock step. Conversely *anti-affinity* rules give a set of VMs that must not be placed on the same host. The DRS load balancing algorithm is not allowed to breach these rules, it does only so in the initial placement scenario.

4 Conclusion

The field of virtualization in IT is in a constant hype due to the continuous appearance of new technologies and challenges generated by them. In this literature study some of the challenges of the management of a virtualized infrastructure were discussed in a very coarse detail. Even at this level it is visible that concerning many challenges of the virtualization at large scale, one cannot hope to find an optimal solution. However, with the introduction of autonomous components, optimization tasks such as resource allocation, become feasible even in large scale, multi site environments. It has also been shown, that in some aspects of backup and recovery related tasks, virtualization enables the blurring of the distinction between system administrators and end users. In the foreseeable future it seems that virtualization will continue to spread, further widening the gap between the hardware and the actual user applications.

5 References

[1] Z. Micskei, D. Tóth *"Bevezető, virtualizációs technológiák áttekinése"* PDF, 2012.09, available: http://www.inf.mit.bme.hu/sites/default/files/materials/category/kateg%C3%B3ria/oktat%C3%A1s/v%C3%A1laszthat%C3%B3-t%C3%A1rgyak/virtualiz%C3%A1ci%C3%B3s-technol%C3%B3gi%C3%A1k-%C3%A9s-alkalmaz%C3%A1saik/12/01-virttech-2012-Bevezeto.pdf

[2] Vijayaraghavan Soundararajan , Kinshuk Govil, *"Challenges in building scalable virtualized datacenter management"*, ACM SIGOPS Operating Systems Review, v.44 n.4, December 2010 doi:10.1145/1899928.1899941

[3] J. Hunter (VMware), *"VMware vSphere Data Protection"*, 2012, PDF, available: http://www.VMware.com/files/pdf/techpaper/Introduction-to-Data-Protection.pdf

[4] Ajay Gulati, Anne Holler, Minwen Ji, Ganesha Shanmuganathan, Carl Waldspurger, Xiaoyun Zhu *"VMware Distributed Resource Management: Design, Implementation and Lessons Learned"*,2012, PDF, available: http://labs.VMware.com/publications/gulati-vmtj-spring2012